AD743171

DNA 2860T

SAI-72-511-LJ

May 1972

THE MORSE CODE WITH COMBINATORIAL GEOMETRY

TOPICAL REPORT

1 January 1971 through 28 February 1972

by

E. A. Straker
W. H. Scott, Jr.
N. R. Byrn

HEADQUARTERS
Defense Nuclear Agency
Washington, D. C. 20305

Contract DASA01-71-C-0044

DDC
RECEIVED
JUN 12 1972
D

Approved for public release; distribution unlimited.

SCIENCE APPLICATIONS, INCORPORATED

P. O. Box 2351, 1250 Prospect St., La Jolla, Calif. 92037 (714) 459-0211

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
Springfield, Va. 22151

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Science Applications, Inc. P. O. Box 2351, 1250 Prospect Street La Jolla, California 92037	Unclassified 2b. GROUP

3 REPORT TITLE

THE MØRSE CODE WITH COMBINATORIAL GEOMETRY

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)

Topical Report, 1 January 1971 through 28 February 1972

5 AUTHOR(S) (First name, middle initial, last name)

Edward A. Straker, William H. Scott, Jr., N. Rick Byrn

6 REPORT DATE	7a. TOTAL NO OF PAGES	7b. NO OF REFS
May 1972	186	12

8a. CONTRACT OR GRANT NO	9a. ORIGINATOR'S REPORT NUMBER(S)
DASA01-71-C-0044 b. PROJECT NO NWER c Task & Subtask XAXPE072 d. Work Unit 04	SAI-72-511-LJ 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) DNA 2860T

10 DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11 SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Director Defense Nuclear Agency Washington, D.C. 20305

13 ABSTRACT

The combinatorial geometry allows efficient Monte Carlo radiation transport calculations of detailed three-dimensional geometries. Because regions and media are formed by combination of basic bodies such as boxes, spheres, cylinders and others, the input required of the user is both relatively simple and easily modified. The MØRSE code is a multigroup neutron and gamma ray transport Monte Carlo code that may solve either neutron, gamma ray, or coupled neutron-gamma ray problems in either the forward and adjoint mode. MØRSE has a wide variety of available input options, including splitting, Russian roulette, exponential transform, energy biasing, importance regions, albedo surfaces, and the scoring options available in the SAMBØ analysis package.

This document details the incorporation of a new version of combinatorial geometry into the MØRSE code and is meant as a user's manual. All new subroutines, including the combinatorial geometry, are documented in detail, including flow charts. Input instructions for all MØRSE options and layouts of all MØRSE commons are given for user reference. The PICTURE code has been modified to work with the combinatorial geometry, allowing two-dimensional printer pictures of combinatorial geometry models. Two MØRSE sample problems and a PICTURE sample problem are discussed. Input and output from these problems and the complete code are available from the Radiation Shielding Information Center.

DD FORM 1 NOV 68 1473

UNCLASSIFIED
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
MØRSE						
combinatorial geometry						
Monte Carlo radiation transport						
neutron and gamma ray						
multigroup cross sections						
three-dimensional geometry						
user's manual						

ABSTRACT

The combinatorial geometry allows efficient Monte Carlo radiation transport calculations of detailed three-dimensional geometries. Because regions and media are formed by combination of basic bodies such as boxes, spheres, cylinders and others, the input required of the user is both relatively simple and easily modified. The MØRSE code is a multigroup neutron and gamma ray transport Monte Carlo code that may solve either neutron, gamma ray, or coupled neutron-gamma ray problems in either the forward or adjoint mode. MØRSE has a wide variety of available input options, including splitting, Russian roulette, exponential transform, energy biasing, importance regions, albedo surfaces, and the scoring options available in the SAMBØ analysis package.

This document details the incorporation of a new version of combinatorial geometry into the MØRSE code and is meant as a user's manual. All new subroutines, including the combinatorial geometry, are documented in detail, including flow charts. Input instructions for all MØRSE options and layouts of all MØRSE commons are given for user reference. The PICTURE code has been modified to work with the combinatorial geometry, allowing two-dimensional printer pictures of combinatorial geometry models. Two MØRSE sample problems and a PICTURE sample problem are discussed. Input and output from these problems and the complete code are available from the Radiation Shielding Information Center.

Preceding page blank

TABLE OF CONTENTS

		Page
I.	INTRODUCTION	1
II.	COMBINATORIAL GEOMETRY	5
	2.1 Body Types	5
	2.2 Subroutines	15
	2.2.1 Subroutine G1	15
	2.2.2 Subroutine LØØKZ	21
	2.2.3 Subroutine GG	24
	2.2.4 Subroutine GENI	27
	2.2.5 Subroutine ALBERT	29
	2.2.6 Subroutine PR	31
	2.2.7 Subroutine NØRML	33
	2.2.8 Subroutine GTVLIN	35
	2.2.9 Subroutine JØMIN	37
	2.2.10 Subroutine GTVØL	40
	2.2.11 Subroutine GØMST	42
	2.2.12 Subroutine EUCLID	45
	2.2.13 Subroutine ENDRUN	48
III.	MODIFICATIONS FOR UNIVAC-1108	55
	3.1 Built-In Fluence Estimators	56

Preceding page blank

TABLE OF CONTENTS (Cont'd.)

		Page
3.2	Collision Density Fluence Estimator	56
3.3	Track Length per Unit Volume Estimator	57
IV.	THE PICTURE PROGRAM	101
4.1	Program PICTURE	102
4.2	Subroutine PRINT	104
4.3	Subroutine MESH	106
V.	MØRSE SAMPLE PROBLEMS	109
5.1	Sample Problem Number 1	109
5.1.1	Subroutine INSCØR - Sample Problem Number 1	111
5.1.2	Subroutine RELCØL - Sample Problem Number 1	113
5.1.3	Subroutine BDRYX - Sample Problem Number 1	115
5.2	Sample Problem Number 2	117
5.3	Sample Problem Number 3	118
APPENDIX A--ENERGY INDEXING SCHEME		123
APPENDIX B--INPUT INSTRUCTIONS		129
B.1	MØRSE Input Instructions	129
B.2	Ø5R Geometry Input Instructions	138
B.2.1	Spherical GEØM	138
B.2.2	Slab GEØM	138

TABLE OF CONTENTS (Cont'd.)

		Page
	B.2.3 Cylindrical GEØM	139
	B.2.4 General GEØM	140
B.3	Combinatorial Geometry Input Instructions	147
B.4	MØRSEC--Cross-Section Module Input Instructions	152
B.5	SAMBØ Analysis Input Instructions	156
B.6	Input to PICTURE	159
REFERENCES	. . . . .	163

LIST OF FIGURES

Figure		Page
1	Examples of combinatorial geometry method	7
2	Use of ØR operators	9
3	Rectangular Parallelepiped (RPP)	10
4	Sphere (SPH)	10
5	Right Circular Cylinder (RCC)	11
6	Right Elliptical Cylinder (REC)	11
7	Truncated Right Angle Cone (TRC)	12
8	Ellipsoid (ELL)	12
9	Right Angle Wedge (WED)	13
10	Box (BØX)	14
11	Arbitrary Polyhedron (ARB)	14
12	General layout of blank common	58
13	Layout of combinatorial geometry data in blank common	59
14	Detailed layout of the FPD-array in blank common	60
15	Detailed layout of the MA array in blank common	63
16	Layout of permanent cross sections in blank common	65
17	Layout of particle bank in blank common	66
18	Layout of fission bank in blank common	67
19	Layout of user area (analysis) in blank common	69

LIST OF FIGURES (Cont'd.)

Figure		Page
A1	Diagram of energy group structure	124

LIST OF TABLES

Table		Page
1	MØRSE Routines that Depend on Which Geometry Module is Used	2
2	Geometry Module Routines	2
3	Indexing of Random Walk Blank Common Arrays	70
4	Indexing of Analysis Arrays in Blank Common	73
5	Definition of Variables in APØLLØ Common	76
6	Definition of Variables in NUTRØN Common	80
7	Definitions of Variables in USER Common	81
8	Definitions of Variables in Common GØMLØC	82
9	Definitions of Variables in Common LØCSIG	84
10	Definitions of Variables in Common GTSC1	89
11	Definitions of Variables in Common PDET	90
12	Definitions of Variables in Common FISBNK	92
13	Definitions of Variables in Common GEØMC as found in Ø5R Geometry	93
14	Definitions of Variables in Common PAREM as found in Combinatorial Geometry	94
15	Definitions of Variables in Common ØRGI	96
16	Definitions of Random Walk Variables in Blank Common	97
17	Definition of Variables in Common PICT	108
18	Comparison of Results of Sample Problem 1	110

LIST OF TABLES (Cont'd.)

Table		Page
19	Results of Last Flight Estimator Sample Problem.	117
20	Output from PICTURE Sample Problem.	119
A1	Numerical Examples for Various Options with Corresponding Energies Given for Each Case. . . .	125
A2	Values of NQT1, NQT2, and NQT3 for Several Cases	127
B1	Sample Group Input Numbers for Some Representative Problems	135
B2	BANKR Arguments (NCØLLS)	136
B3	Ø5R General Geometry Input Example.	145
B4	Input Required on CGB Cards for Each Body Type.	150

I. INTRODUCTION

The MØRSE Monte Carlo code was first distributed[1] in early 1970 with a revised version[2, 3] released in the fall of 1970. There have been several additional features[4, 8] added in the past year. This report describes the addition of a version of the MAGI developed combinatorial geometry[9] (CG) to the MØRSE code and also serves to document a version which is operational on the UNIVAC 1108. The definition of variables in labelled commons, layout of blank common and input data have been collected for easy reference. A description of the PICTURE program for the combinatorial geometry package and descriptions of routines required to solve two sample problems are included.

The replacement of the Ø5R geometry module with the CG module required significant modification of the CG package. These modifications were required so that the CG module could be added to MØRSE rather than adding MORSE to the CG package. Only those MØRSE routines that have an interface with the geometry package were changed and all options in MØRSE are available with the CG module. To retain all MØRSE options it was necessary to write several new routines as part of the CG module.

Features of MØRSE which are dependent on the geometry module include the ability to (1) determine both the media and an importance region given a set of coordinates, (2) to track a particle through the system including internal voids and surrounded by an external void, (3) determine the number of mean free paths between any two sets of coordinates (one may be in either an internal or an external void), (4) determine the vector normal to a surface at any point, and (5) permit particle reflection at a boundary without "getting lost." Table I lists the MØRSE module routines which were modified to permit an interface

TABLE I.

MØRSE Routines that Depend on Which Geometry Module is Used

EUCLID (3)
GØMST (2)
INPUT1
MSØUR (1)

with the CG module. The numbers after the routines relate the features mentioned above with the function of the routines. The routines without numbers are concerned with the input of geometry information. Table II

TABLE II.

Geometry Module Routines

Ø5R Spherical Geometry		Combinatorial Geometry	
GEØM	(2, 3)	GENI	
GØMFLP	(5)	GTVLIN	
JØMIN		GG	(2, 3)
LØØKZ	(1)	G1	(2, 3)
NØRML	(4)	ALBERT	
GTVØL		PR	
		GØMFLP	(5)
		JØMIN	
		LØØKZ	(1)
		NØRML	(4)
		GTVØL	

lists the geometry module routines for the Ø5R spherical and the combinatorial geometry packages.

Only minor changes were made to INPUT and MSØUR and no additional documentation is provided. INPUT was split into two routines, INPUT1 and INPUT2, to allow a more efficient overlay. Numerous changes were required for subroutine EUCLID and GØMST and new descriptions and flow charts of these routines are given. A complete description of the combinatorial geometry package is included and sample problems are provided.

II. COMBINATORIAL GEOMETRY

2.1 Body Types

Combinatorial geometry (CG) describes general three dimensional material configurations by considering unions, differences intersections of simple bodies such as spheres, boxes, cylinders, etc. In effect, the geometric description subdivides the problem space into unique zones.* Each zone is the result of combining one or more of the following geometric bodies.

1. Rectangular Parallelepiped (RPP)
2. Box (An RPP randomly oriented in space)
3. Sphere
4. Right Circular Cylinder
5. Right Elliptic Cylinder
6. Truncated Right Angle Cone
7. Ellipsoid
8. Right Angle Wedge
9. Arbitrary Convex Polyhedron of 4, 5, or 6 sides.

Body types 8-9 may be arbitrarily oriented with respect to the x, y, z coordinate axes used to determine the space. Body 1, a special body described below, must have sides which are parallel to the coordinate axes.

The basic technique for the description of the geometry consists of defining the location and shape of the various zones in terms of

* To avoid confusion between importance regions and combinatorial geometry regions, we depart from previous combinatorial geometry descriptions and use the term zone to indicate a combinatorial geometry region which is designated by the variable IR. The term region is reserved for an importance region. Thus the zone index is IR.

the intersections and unions of the geometric bodies. A special operator notation involving the symbols (+), (-), and (ØR) is used to describe the intersections and unions. These symbols are used by the program to construct information relating material descriptions to the body definitions.

If a body appears in a zone description with a (+) operator, it means that the zone being described is wholly contained in the body. If a body appears in a zone description with a (-) operator, it means that the zone being described is wholly outside the body. If the body appears with an (ØR) operator, it means that the zone being described includes all points in the body. In some instances, a zone may be described in terms of subzones lumped together by (ØR) statements. When (ØR) operators are used there are always two or more of them, and they refer to all body numbers following them, either (+) or (-).

Techniques for describing a particular geometry are best illustrated by examples. Consider an object composed of a sphere and a cylinder as shown in Fig. 1. To describe the object, one takes a spherical body (2) penetrated by a cylindrical body (3) (see Fig. 1). If the materials in the sphere and cylinder are the same, then they can be considered as one zone, say zone I (Fig. 1c). The description of zone I would be

$$I = \text{ØR} + 2\ \text{ØR} + 3 \quad .$$

This means that a point is in zone I if it is <u>either</u> inside body 2 <u>or</u> inside body 3.

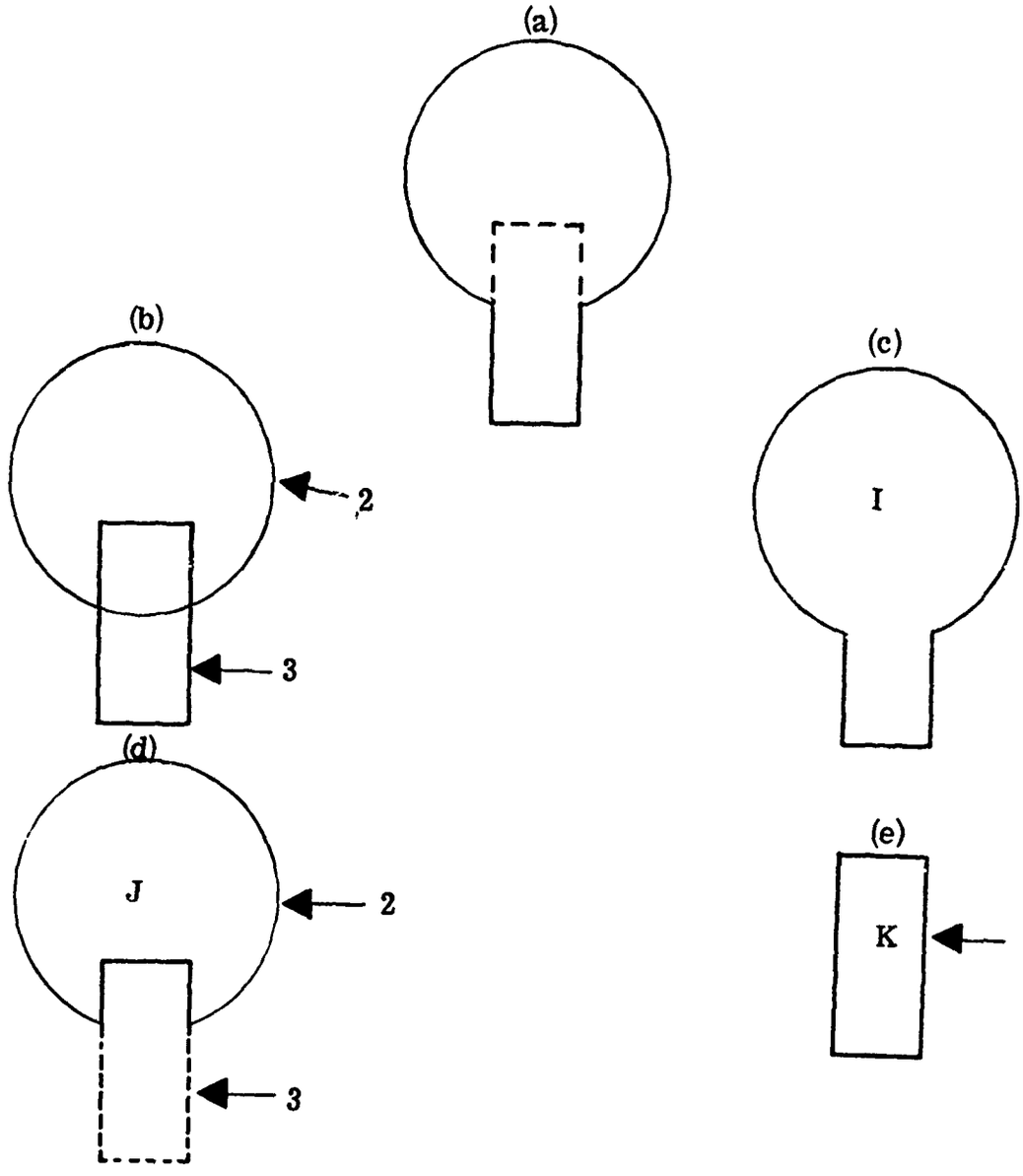


Fig. 1. Examples of combinatorial geometry method.

If different materials are used in the sphere and cylinder, then the sphere with a cylindrical hole in it would be given a different zone number (say J) from that of the cylinder (K).

The description of zone J would be (Fig. 1d):

$$J = +2 -3 \quad .$$

This means that points in zone J are all those points inside body 2 which are not inside body 3.

The description of zone K is simply (Fig. 1e):

$$K = +3 \quad .$$

That is, all points in zone K lie inside body 3.

Combinations of more than two bodies and similar zone descriptions could contain a long string of (+), (-), and (ØR) operators. It is important however to remember that every spatial point in the geometry must be located in one and only one zone.

As a more complicated example of the use of the (ØR) operator, consider the system shown in Fig. 2 consisting of the shaded zone A and the unshaded zone B. These zones can be described by the two BØX's, bodies 1 and 3, and the RCC, body 2. The zone description would be

$$A = +1 +2$$

and

$$B = \text{ØR} +3 -1 \text{ØR} +3 -2 \quad .$$

Notice that the ØR operator refers to all following body numbers until the next ØR operator is reached.

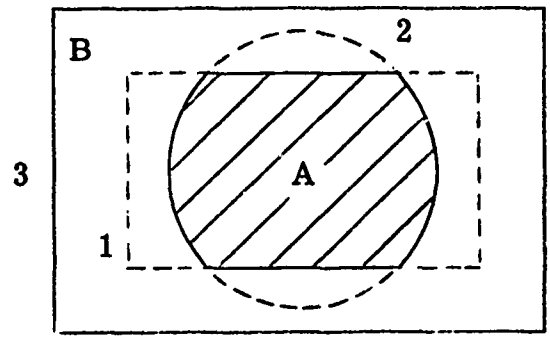


Fig. 2. Use of ØR operators.

The geometry must be specified by establishing two tables. The first table describes the type and location of the set of bodies used in the geometrical description. The second table identifies the physical zones in terms of these bodies. The input routine processes these tables to put the data in the form required for ray tracing. Because the ray tracing routines cannot track across the outermost body, all of the zones must be within a surrounding external void so that all escaping particles are absorbed. Also no point may be in more than one zone.

The information required to specify each type of body is as follows:

a. Rectangular Parallelepiped (RPP)

Specify the minimum and maximum values of the x, y, and z coordinates which bound the parallelepiped.

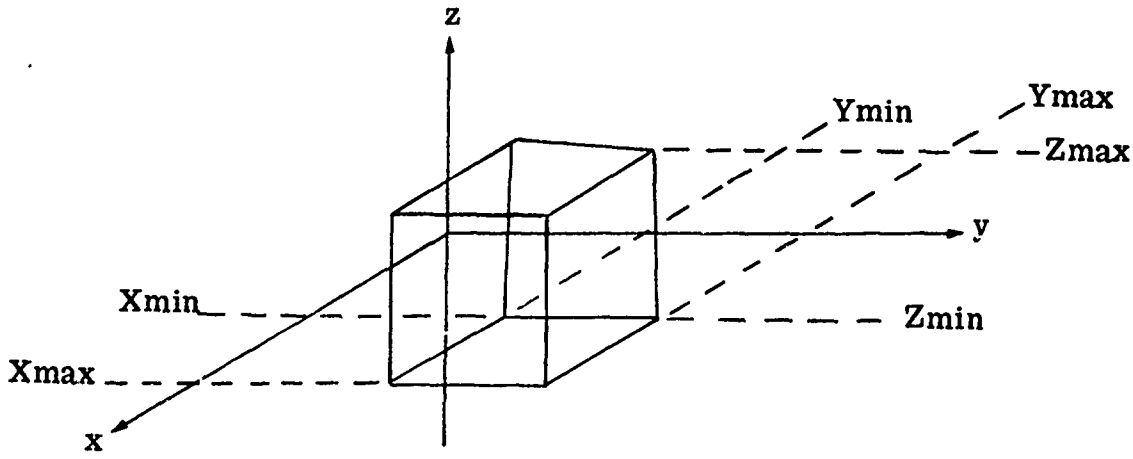


Fig. 3. Rectangular Parallelepiped (RPP).

b) Sphere (SPH)

Specify the vertex $\underline{V}$ at the center and the scalar, R, denoting the radius.

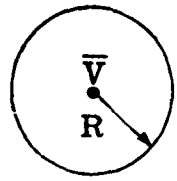


Fig. 4. Sphere (SPH).

c) Right Circular Cylinder (RCC)

Specify the vertex $\underline{V}$ at the center of one base, a height vector, $\underline{H}$, expressed in terms of its x, y, and z components, and a scalar, R, denoting the radius.

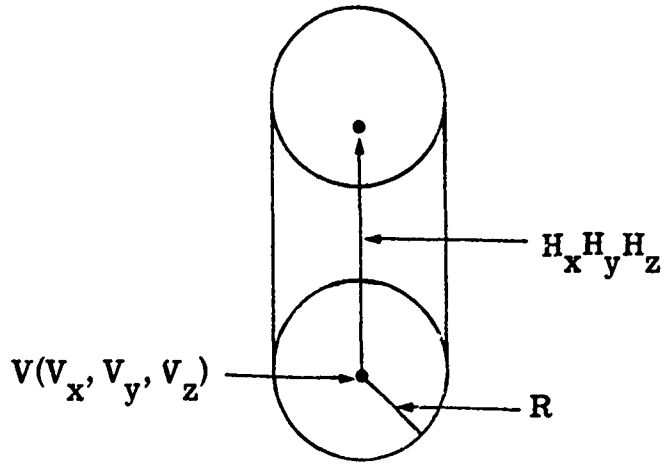


Fig. 5. Right Circular Cylinder (RCC).

d) Right Elliptical Cylinder (REC)

Specify coordinates of the center of the base ellipse, a height vector, and two vectors in the plane of the base defining the major and minor axes. Presently this body is not implemented.

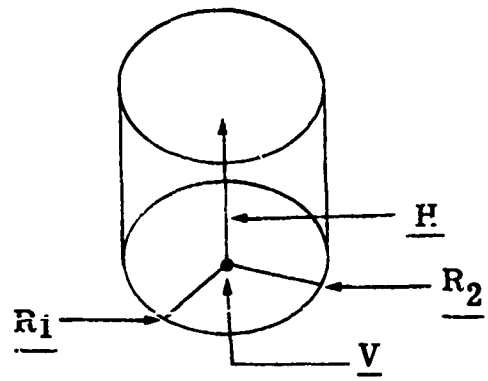


Fig. 6. Right Elliptical Cylinder (REC).

e) Truncated Right Angle Cone (TRC)

Specify a vertex $\underline{V}$ at the center of the lower base, the height vector, $\underline{H}$, expressed in terms of its x, y, z components, and two scalars, R_1 and R_2, denoting the radii of the lower and upper bases.

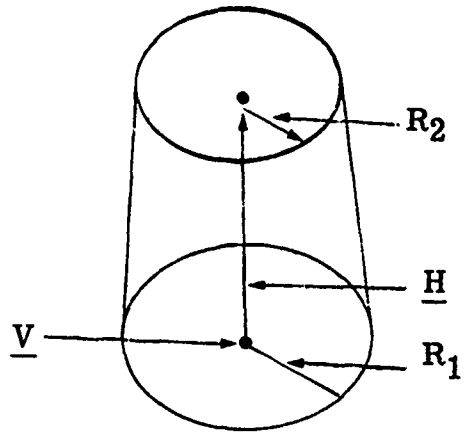


Fig. 7. Truncated Right Angle Cone (TRC).

f) Ellipsoid (ELL)

Specify two vertices, $\underline{V_1}$ and $\underline{V_2}$, denoting the coordinates of the foci and a scalar, R, denoting the length of the major axis.

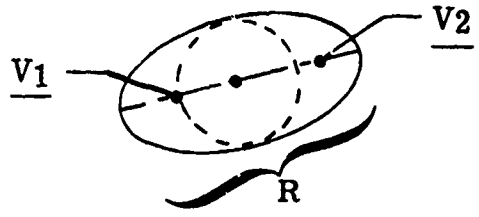


Fig. 8. Ellipsoid (ELL).

g) Wedge (WED)

Specify the vertex $\underline{V}$ at one of the corners by giving its (x, y, z) coordinates. Specify a set of three mutually perpendicular vectors, $\underline{a_i}$, with a_1 and a_2 describing the two legs of the right triangle of the wedge. That is, the x, y, and z components of the height, width, and length vectors are given.

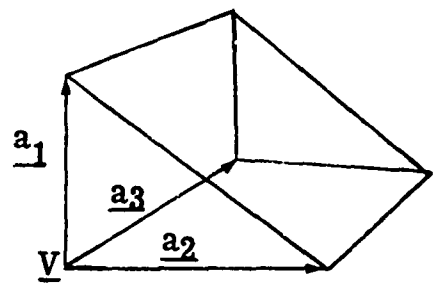


Fig. 9. Right Angle Wedge (WED).

h) Box (BØX)

Specify the vertex $\underline{V}$ at one of the corners by giving its (x, y, z) coordinates. Specify a set of three mutually perpendicular vectors, $\underline{a_i}$ representing the height, width, and length of the box, respectively. That is, the x, y, and z components of the height, width, and length vectors are given.

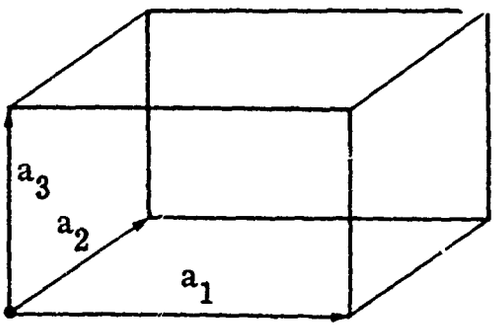


Fig. 10. Box (BØX).

i) Arbitrary Polyhedron (ARB)

Assign an index (1 to 8) to each vertex. For each vertex, give the x, y, z coordinates. Each of the six faces are then described by a four-digit number giving the indices of the four vertex points in that face. For each face these indices must be entered in either clockwise or counterclockwise order.

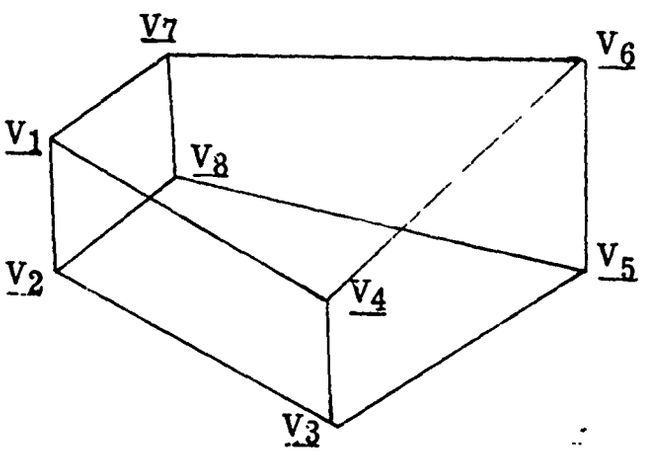


Fig. 11. Arbitrary Polyhedron (ARB).

Detailed input instructions are given in Appendix B.

2.2 Subroutines

Descriptions of the combinatorial geometry routines are given with logical flow charts depicting their functions. Two MØRSE routines, GØMST and LØØKZ required significant modifications for use with the CG package and descriptions of these routines are also provided.

2.2.1 Subroutine G1(S, MA, FPD, LØCREG, NUMBØD, IRØR, IR1, IR2)

G1 is the control routine for the combinatorial geometry. On one call, it calculates the distance travelled in the present zone, and the number IR of the next zone to be entered. Essentially GG is called for each body adjacent to the present zone, calculating RIN and RØUT, the distances to entry and exit of the body along the trajectory. The next zone to be entered is determined by again calling GG to calculate RIN and RØUT for each body adjacent to the next possible zone. These next possible zones are determined by examining a list of all the previous zones entered on crossing this body. RIN and RØUT are checked against the input zone descriptions to determine the correct zone. If it is not found in the list of previous zones, all other zones are examined in a similar fashion, and when the correct zone is found, it is added to the list of previous zones for that body. If the new zone is different from the old, G1 returns; otherwise G1 continues tracking until a different zone is encountered. One change added to the MØRSE version of G1 is that if the distance to the next boundary is greater than the distance to scattering, G1 returns without determining the next zone past the boundary, setting the flag MARKG in common ØRGI.

Called from: GØMST, EUCLID, MESH
Subroutines called: GG
Commons required: PAREM, GØMLØC, DBG, ØRGI

Variables required:

XB(3)	-	starting coordinates of present trajectory,
WB(3)	-	direction cosines of trajectory,
IR	-	present zone,
DIST	-	present distance from XB(3),
DIST0	-	distance from XB(3) to next scattering point,
NASC	-	less than zero if this is a new trajectory,
KLØØP	-	trajectory index,
PINF	-	machine infinity,
MA, FPD, LØCREG, NUMBØD, IRØR, IR1, IR2	-	locations in blank common used for variable dimensioning.

Variables changed:

KLØØP	-	trajectory index incremented if this is a new trajectory,
NASC	-	next body intersected by trajectory,
LSURF	-	surface of body NASC crossed at next intersection (negative if leaving and positive if entering NASC),
DIST	-	distance from XB(3) to next intersection or collision site,
MARK	-	set to 1 if distance to collision (DIST0) is less than distance to next intersection (otherwise 0),
S	-	distance travelled on this call to G1,
IRPRIM	-	zone to be entered on boundary crossing,
MA(INEXT)	-	new zone added to the list of next possible zones for body NBØ,
MA(INEX)	-	location in MA of next item in the next possible zone list for body NBØ (these lists leap-frog through the end of the MA array).

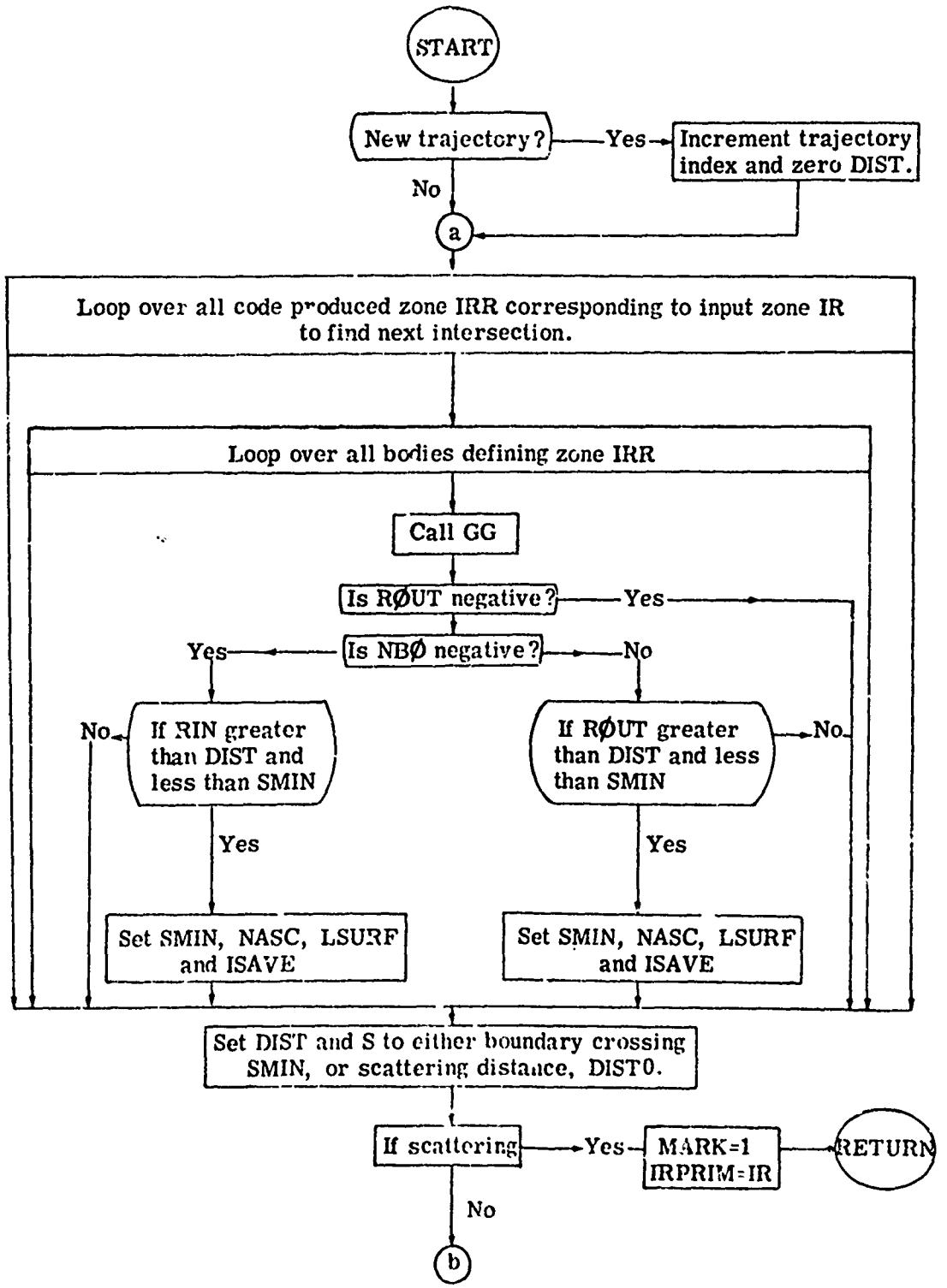
Significant internal variables:

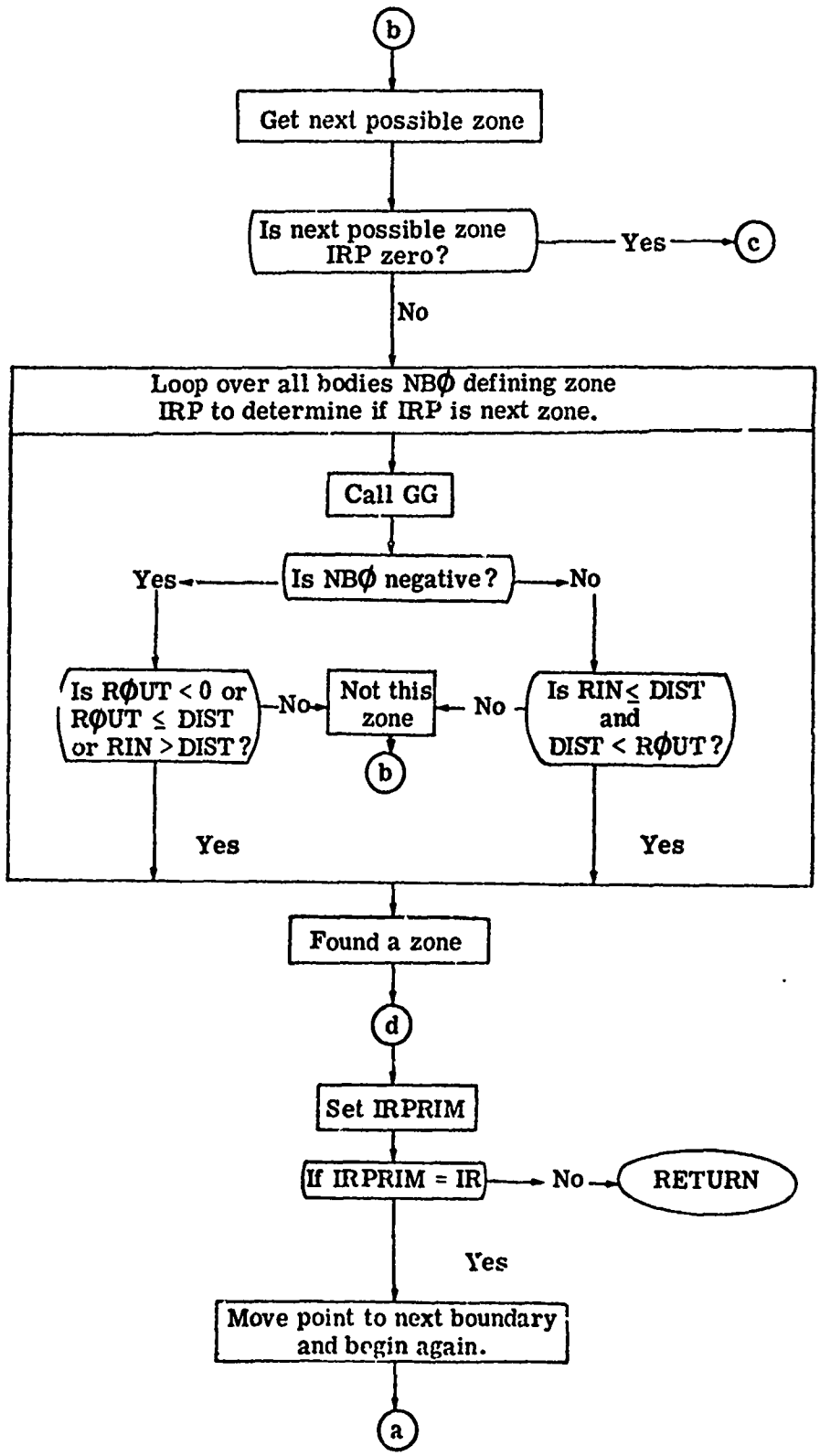
NBØ	-	absolute value is body being considered while a negative or positive sign indicates that zone IR or IRP is outside or inside the body respectively,
RØUT	-	distance to exit of body NBØ calculated by GG,
RIN	-	distance to entry of body NBØ calculated by GG,
LRI	-	surface of body NBØ entered by trajectory,
LRØ	-	surface of body NBØ trajectory exits,
IRP	-	zone being considered as next zone.

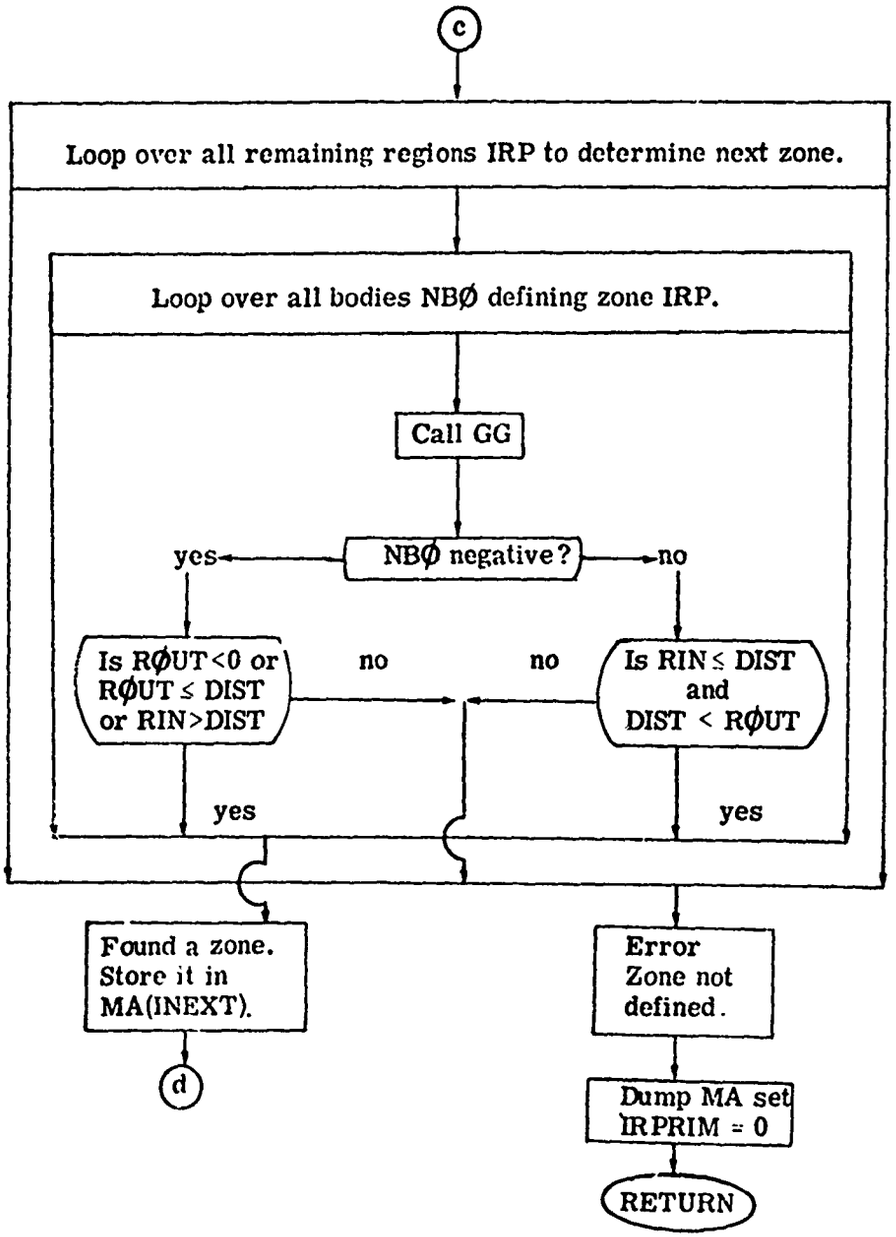
Limitations:

IØUT	-	output logical unit is set to 6 in data statement.

Subroutine G1 (S, MA, FPD, LØCREG, NUMBØD, IRØR, IR1, IR2)




b
Get next possible zone
Is next possible zone IRP zero?
Yes
c
No
Loop over all bodies NBØ defining zone IRP to determine if IRP is next zone.
Call GG
Is NBØ negative?
Yes
No
Is RØUT < 0 or RØUT ≤ DIST or RIN > DIST?
No
Not this zone
No
Is RIN≤ DIST and DIST < RØUT?
b
Yes
Yes
Found a zone
d
Set IRPRIM
If IRPRIM = IR
No
RETURN
Yes
Move point to next boundary and begin again.
a


c
Loop over all remaining regions IRP to determine next zone.
Loop over all bodies NBØ defining zone IRP.
Call GG
NBØ negative?
yes
no
Is RØUT<0 or RØUT ≤ DIST or RIN>DIST
no
no
Is RIN ≤ DIST and DIST < RØUT
yes
yes
Found a zone. Store it in MA(INEXT).
d
Error Zone not defined.
Dump MA set IRPRIM = 0
RETURN

2.2.2 Subroutine LØØKZ (X, Y, Z, MA, FPD, LØCREG, NUMBØD, IRØR, NSØR)*

The purpose of this routine is to return the combinatorial geometry zone of point (X, Y, Z) so that tracking can be initialized. The coding has been borrowed from the second half of subroutine G1 and adapted to determine the zone of a source particle. For efficiency LØØKZ builds a list of possible source zones to search on future calls. If the region is not found on this list, all other zones are examined and upon determining the new source zone, it too is added to the list. Notice that the starting direction cosines (.8, .6, 0.0) are assumed in LØØKZ, but may be changed elsewhere.

Routines called: GG

Commons required: ØRGI, PAREM, GØMLØC, and DBG.

Variables required:

X, Y, Z - coordinates for which a zone is desired,

MA, FPD, LØCREG, NUMBØD, IRØR, NSØR - locations in blank common used for variable dimensioning.

Varaibles changed:

KLØØP - trajectory index is incremented,

NMED - common ØRGI variable set to correct zone number,

NSØR(INEXT) - new zone added to list of possible source zones.

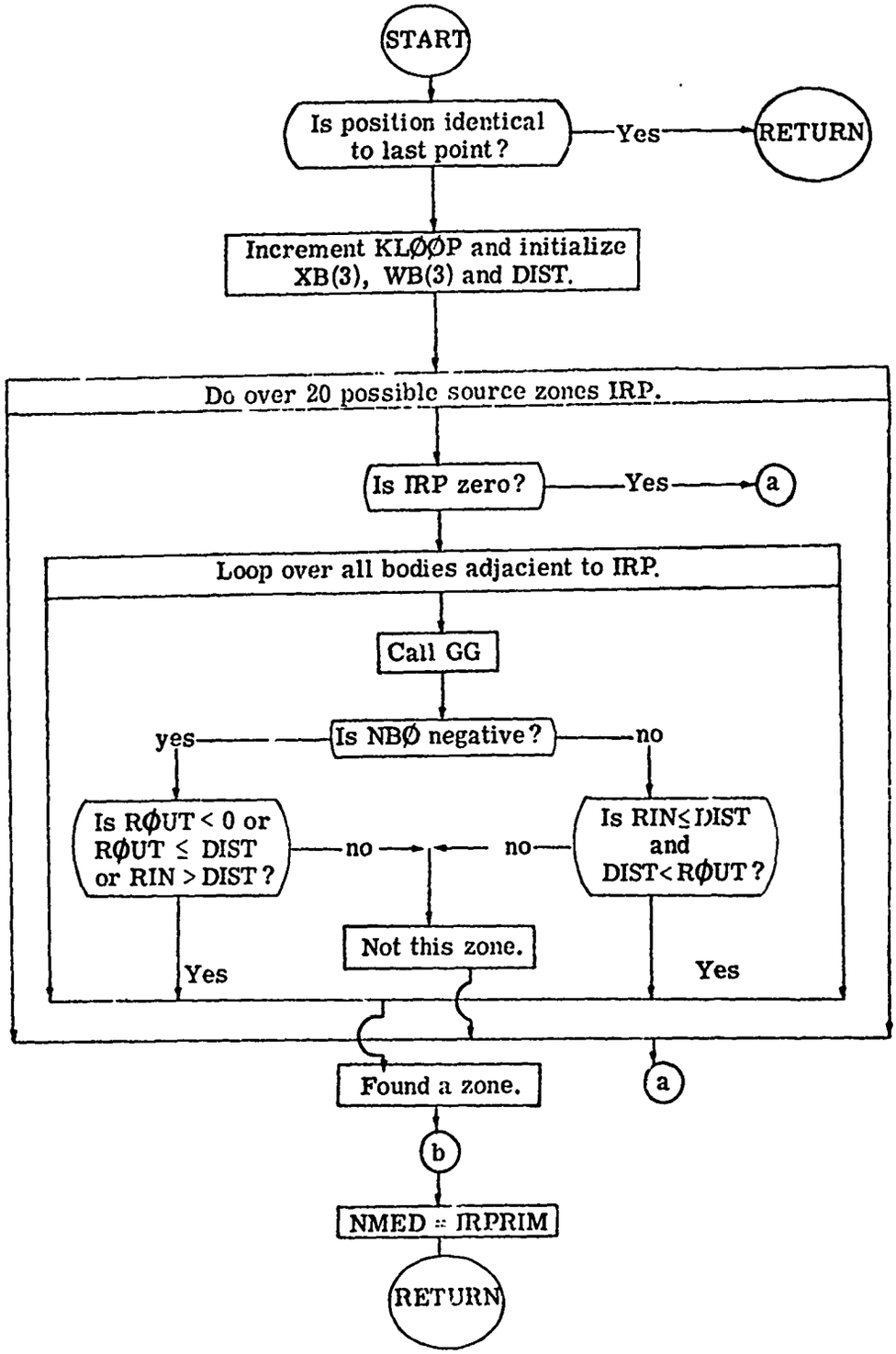
Significant internal variables:

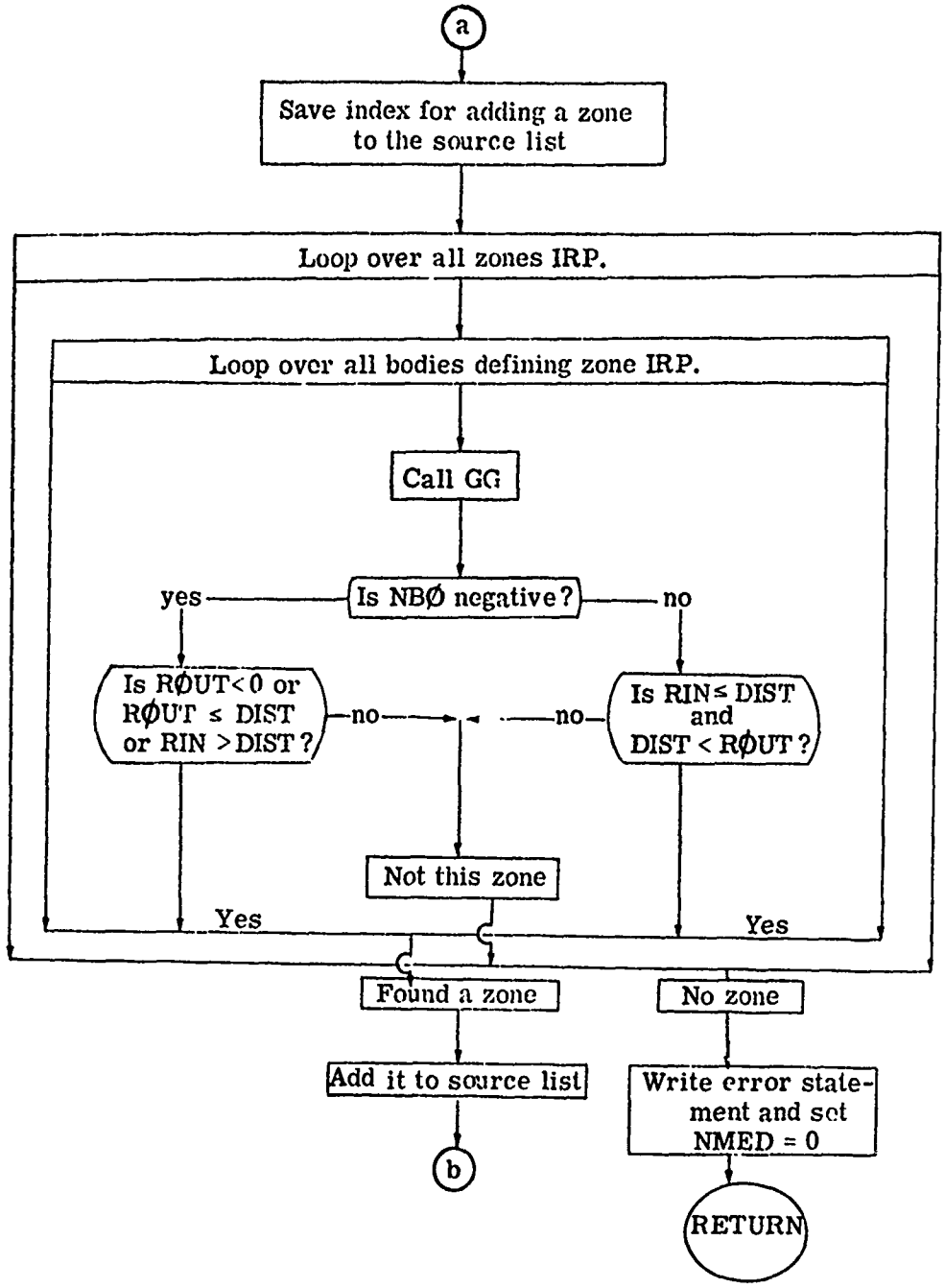
WB(3) - set to .8, .6, 0.0 so that LØØKZ need not be called with a direction

---

* CG version.

Subroutine LØØKZ (X, Y, Z, MA, FPD, LØCREG, NUMBØD, IRØR, NSØR)




a
Save index for adding a zone to the source list
Loop over all zones IRP.
Loop over all bodies defining zone IRP.
Call GG
Is NBØ negative?
yes
no
Is RØUT<0 or RØUT ≤ DIST or RIN >DIST?
no
no
Is RIN≤ DIST and DIST < RØUT?
Not this zone
Yes
Yes
Found a zone
No zone
Add it to source list
Write error statement and set NMED = 0
b
RETURN

2.2.3 Subroutine GG (LØCAT, MA, FPD)

GG is the workhorse of the combinatorial geometry, computing distances to intersections for all body types. It is called from G1 or LØØKZ to compute distance to entry and distance to exit to a body whose location in the MA array is flagged by the argument LØCAT. Each time a distance to entry, RIN, and a distance to exit, RØUT, are calculated for a body they are stored in the MA array together with LRI and LRØ, the indices of the entry and exit surfaces of the body. Also stored at this time is KLØØP, the particles trajectory index. On a subsequent call to GG for that body, KLØØP is checked against the earlier value, now LØØP. If they are the same, the old values of RIN, RØUT, LRI, and LRØ are retrieved so that GG can return immediately.

If it is necessary to compute a new trajectory a different area of coding is entered for each body type to calculate RIN, RØUT, LRI and LRØ.

Called from: G1 and LØØKZ

Subroutines called: none

Commons required: PAREM, GEØM2

Variables required:

LOCAT	-	starting location in the MA array of integer data for the appropriate body,
KLØØP	-	trajectory index,
PINF	-	machine infinity (stored in RIN and RØUT when the trajectory misses the body),
MA, FPD	-	locations in blank common required for variable dimensioning.

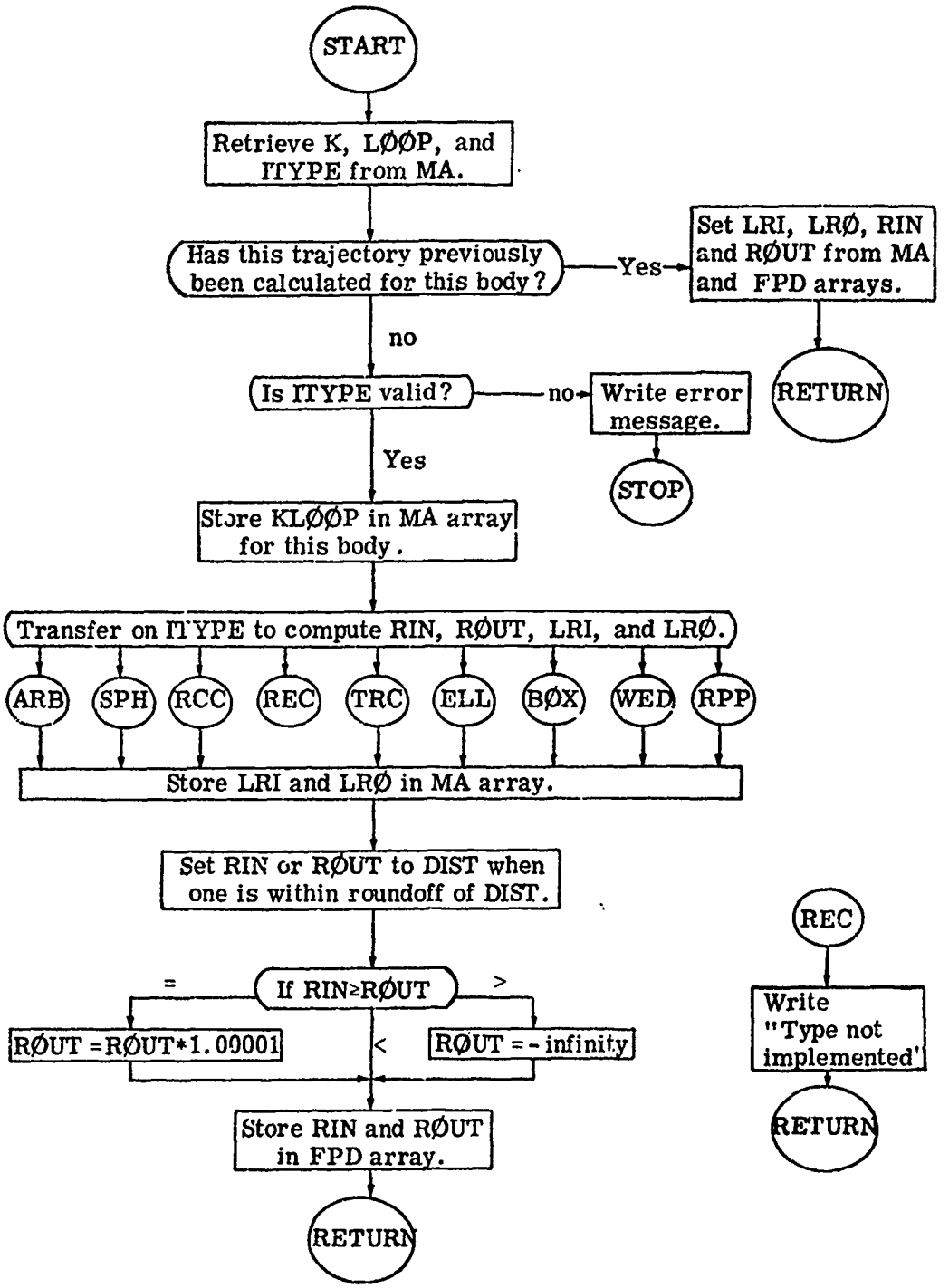
Variables changed:

RIN	-	distance to entrance,
RØUT	-	distance to exit,
LRI	-	surface of entrance
LRØ	-	surface of exit.

Significant internal variables:

L	-	same as LØCAT, starting location in MA array of body data for the appropriate body,
K	-	starting location in FPD array of floating point data for the appropriate body.

Subroutine GG (LØCAT, MA, FPD)



2.2.4 Subroutine GENI (MA, FPD, LØCREG, NUMBØD, IRØR, MRIZ, MRCZ, MMIZ, MMCZ, KR1, KR2, I1, IØUT, IN)

Although subroutine JØMIN is used to read in all of the combinatorial geometry data except the region volumes, that data is put into the proper storage location in blank common by GENI. This is accomplished by having JØMIN write the body and zone data on a mass storage unit, so that GENI can retrieve that data. GENI assigns the data to blank common in the area set aside by JØMIN. For an arbitrary polyhedron body, GENI calls ALBERT to handle a portion of the data. In addition, GENI outputs the input information concerning the geometry data. GENI computes certain geometry dependent data needed during the random walk.

Called from: JØMIN

Subroutines called: ALBERT

Commons required: Blank, GØMLØC, PAREM

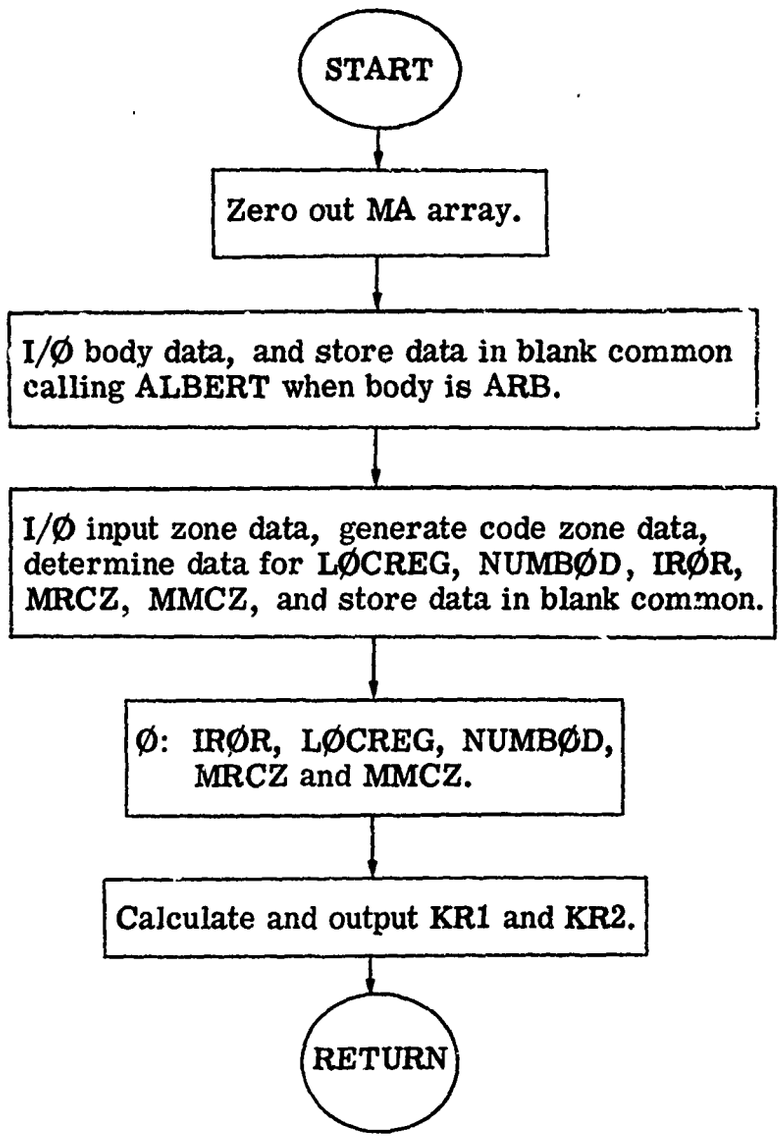
Variables required:

I1	-	input unit,
IØUT	-	output unit,
IN	-	bulk storage logical unit.

Variables input: Geometry data for zones and bodies is retrieved from mass storage unit IN.

Variables changed: MA(I), FPD(I), LØCREG(I), IRØR(I), MRCZ(I), MMCZ(I), KR2(I)--see Fig. 13, layout of combinatorial geometry storage in blank common.

Subroutine GENI (MA, FPD, LØCREG, NUMBØD, IRØR, MRIZ, MRCZ, MMIZ, MMCZ, KR1, KR2, IN)



2.2.5 Subroutine ALBERT (F, IERR)

ALBERT is called by subroutine GENI to process the arbitrary polyhedron (ARB) body data before storage in the FPD array. The ARB body data as read by GENI consists of the coordinates of all corners (eight for a six-sided figure), followed by a packed decimal number for each side indicating which corner points make up that side. ALBERT processes this data, returning a unit normal vector and a minimum distance to origin for each plane containing a side of the ARB. An ARB can have up to six sides. These unit vectors and distances then replace the original ARB data in the FPD array. The number of sides and a distance that is characteristic of the ARB's minimum dimension are also stored in the FPD array. This minimum distance is later used for round-off tests.

Called from: GENI

Subroutines called: none

Commons required: none

Variables required:

F	-	FPD array read from GENI

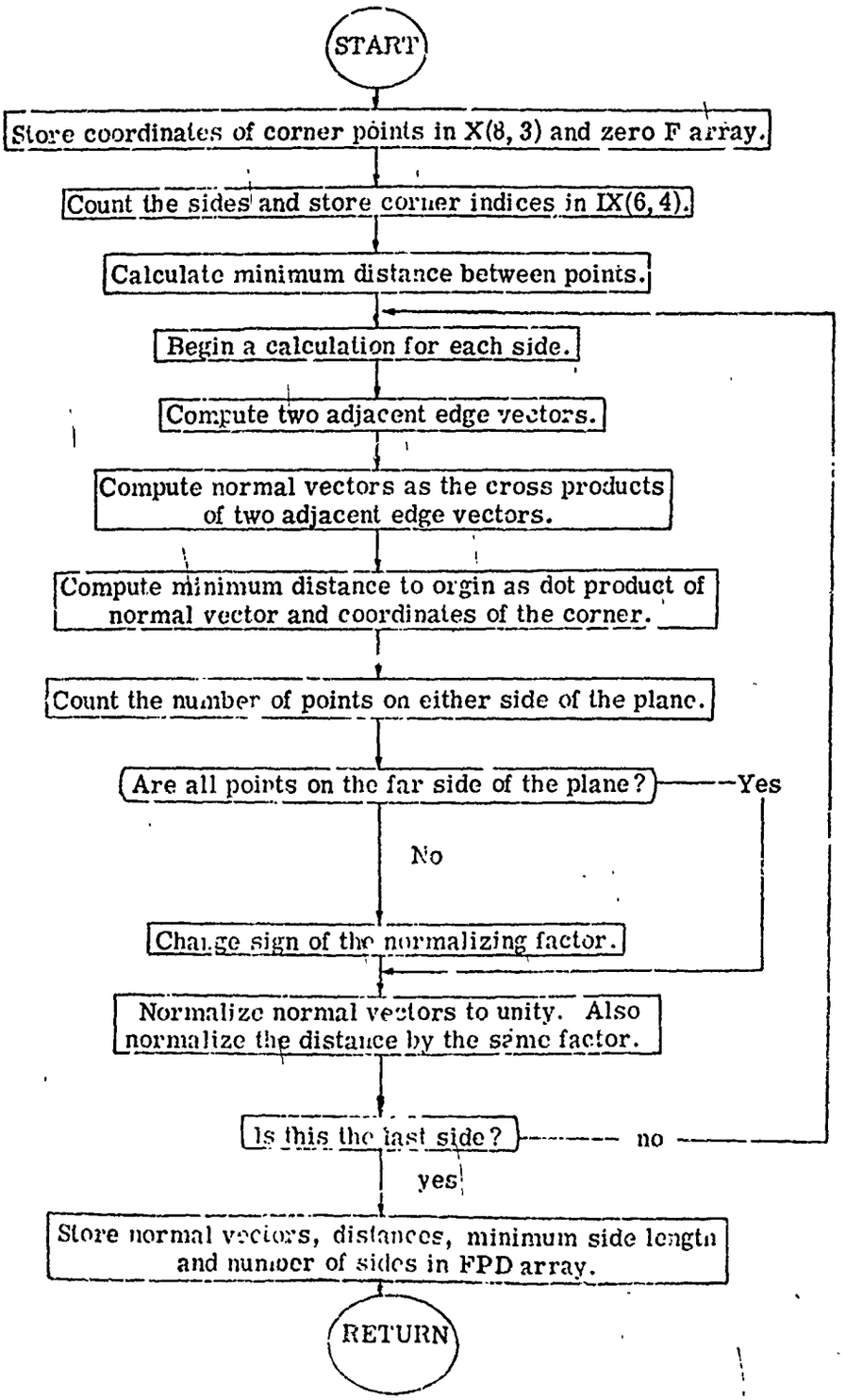
Variables changed:

F	-	FPD array changed to contain unit normals and distances to origin for each plane. Number of sides and a minimum distance are also stored.

Significant internal variables:

X(8, 3)	-	coordinates of corner points,
IS(6, 4)	-	indicates corners contained in each of the possible six surfaces,
V(3, 4)	-	edge vectors for a given surface (an inner cross product then gives the normal vector).

Subroutine ALBERT (F, IERR)

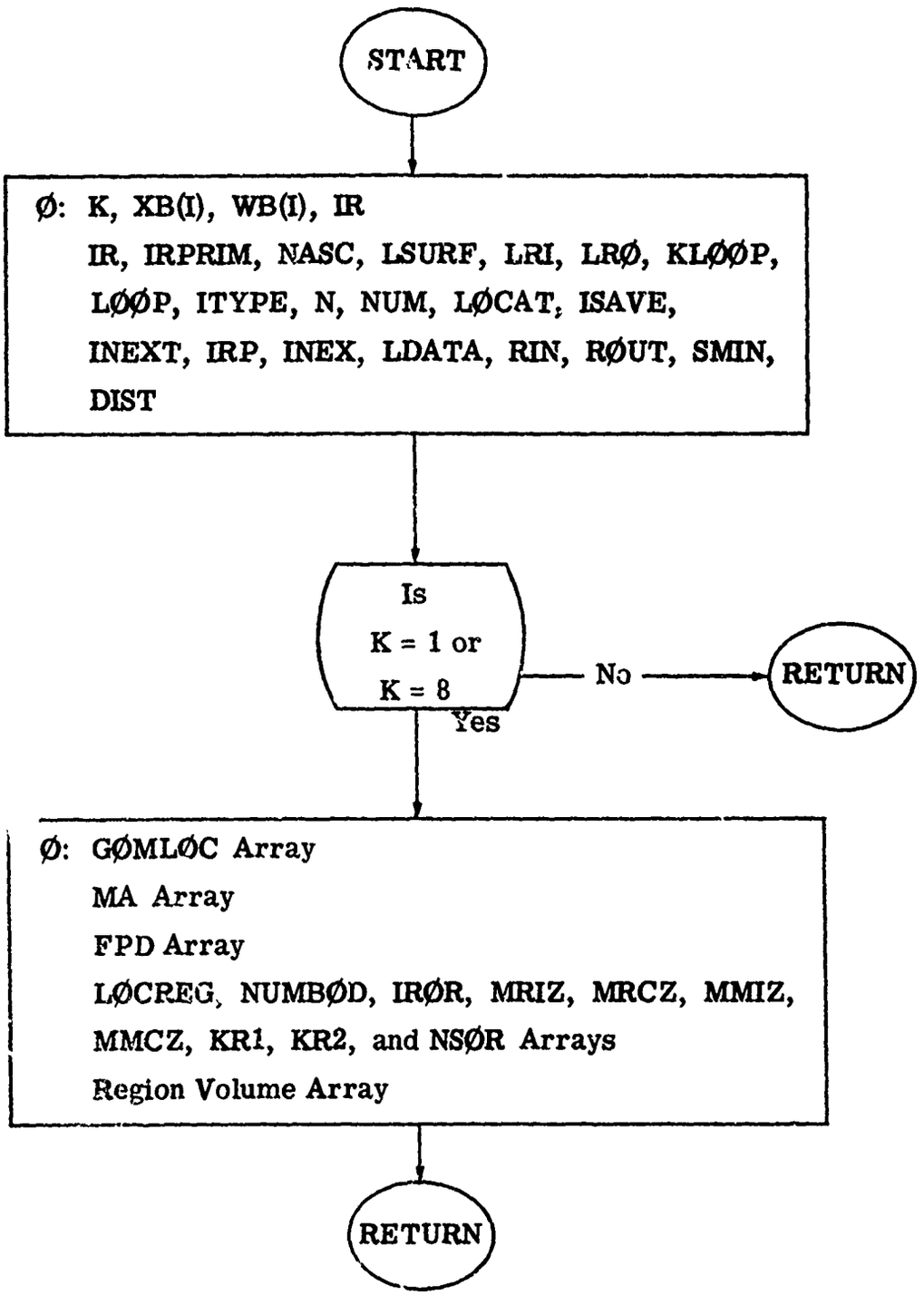


2.2.6 Subroutine PR (Combinatorial Geometry)

Subroutine PR is called from various locations in the combinatorial geometry package (GENI, G1, LØØKZ and NØRML) whenever intermediate or debugging output is required. The amount of geometry data which is output depends on the value of the argument. If this argument is 1 or 8, all of the geometry data in blank common is printed, otherwise only selected variables are output. By comparing the argument value given in the output with the source listing, the geometry data at a given time in the execution can be determined. The call to PR is initiated by setting the IDBG variable to a nonzero value. Because of the large amount of output generated, this option should not be used during a normal execution.

Called from: GENI, G1, LØØKZ, NØRML
Subroutine called: none
Commons required: blank, PAREM, GØMLØC, DBG

Subroutine PR (K)



2.2.7 Subroutine NØRML (MA, FPD, LØCREG, NUMBØD)*

The purpose of subroutine NØRML is to return a unit vector to a combinatorial geometry body NASC at point $\overrightarrow{XB} + D * \overrightarrow{WB}$ which must be on the surface LSURF of body NASC. This unit vector is useful either for albedo scattering or boundary crossing flux estimates. The sign of the unit vector is chosen so that $\overrightarrow{WB} \cdot \hat{n}$ is negative meaning that the unit vector will point against the particle direction.

Called from: ALBDØ

Commons required: NØRMAL, GØMLØC, PAREM

Variables required:

NASC - body number particle is on,

LSURF - surface of body NASC,

XB(3), WB(3), DIST - trajectory parameters,

MA, FPD, LØCREG, NUMBØD - locations in blank common used for variable dimensioning.

Variables changed:

UN(3) - direction cosines of unit vector stored in common NØRMAL.

Significant internal variables:

XP(3) - set to $\overrightarrow{XB}$ + DIST * $\overrightarrow{WB}$,

X(3) - usually used to relate the intersection point to a body centered coordinate system,

H(3) - usually used as a body orientation,

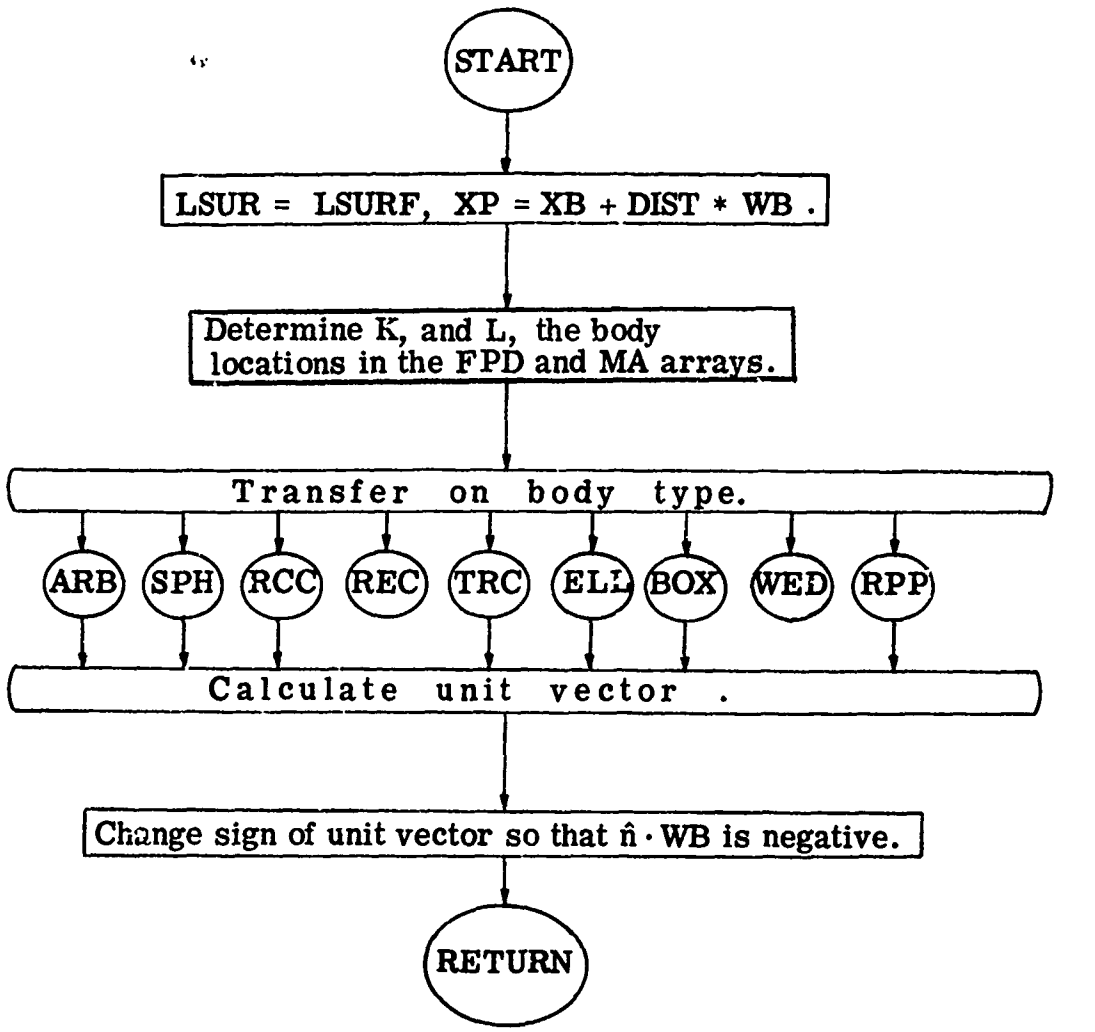
L - body location in MA array,

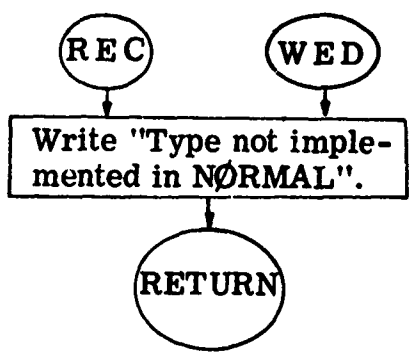
K - body location in FPD array.

---

* CG version.

Subroutine NØRML (MA, FPD, LØCREG, NUMBØD)





2.2.8 Subroutine CVLIN (FPD, MRIZ, VNØR, IVØPT, NIR, NIZ, I1, I0,

This routine is called by JØMIN to read in or to calculate the volume of each region in the geometry. The four options available are (1) to set each volume to 1., (2) to calculate the volumes for concentric spheres, (3) to calculate the volumes for slabs (not coded at present), or (4) to read in the volumes for each region from cards. The volumes are stored in blank common and are only used by the track length and collision density estimators.

Called from: JØMIN

Subroutines called: none

Commons required: none

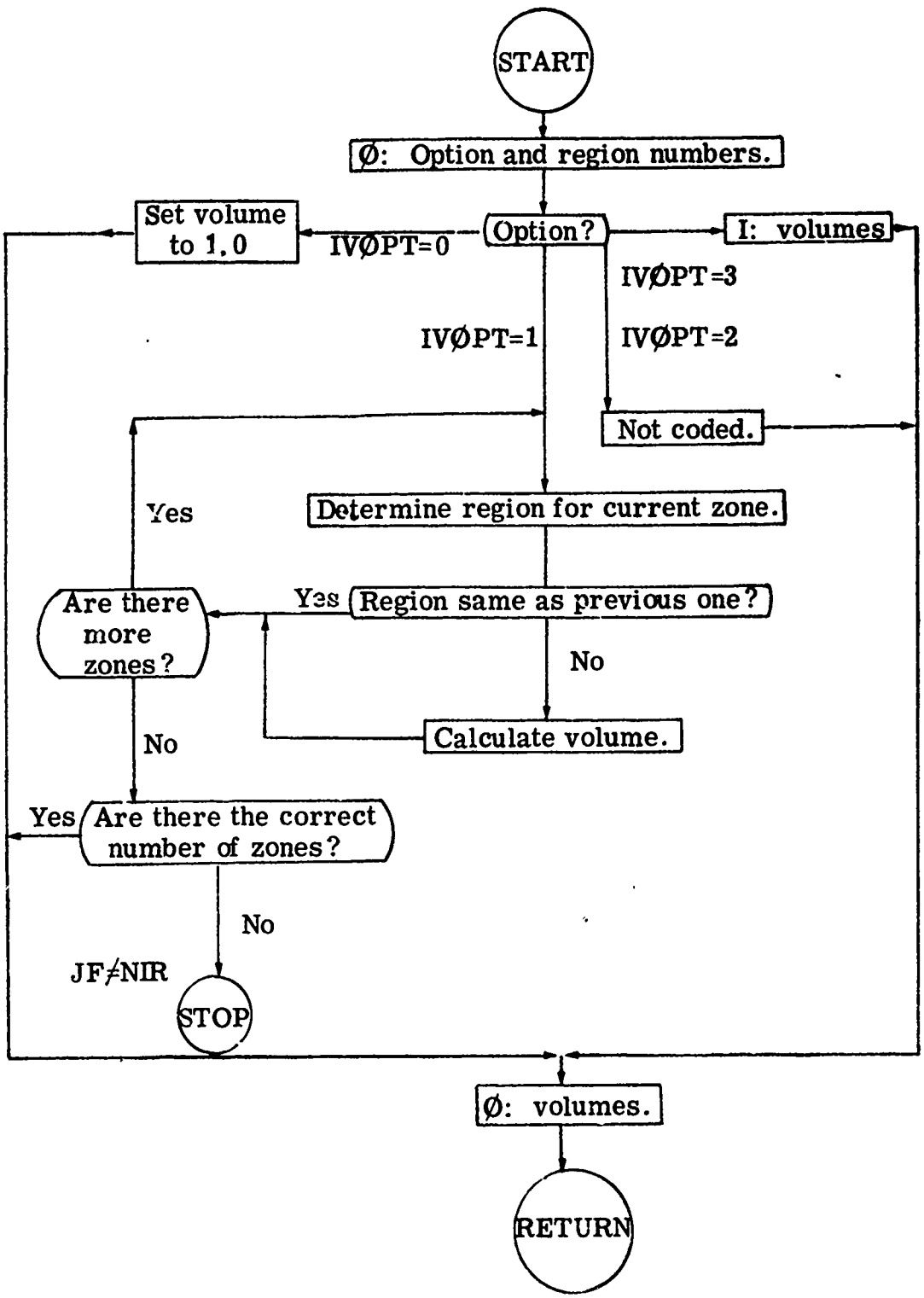
Variables required:

FPD	-	array containing zone description data,
MRIZ	-	array which relates MØRSE region to input zones,
VNØR	-	array containing the MØRSE volumes,
IVØPT	-	options for determining volumes,
NIR	-	number of regions,
NIZ	-	number of zones,
I1, I0	-	input and output logical units.

Variables changed:

Variables in blank common starting at KVØL.

Subroutine GTVLIN (FPD, MRIZ, VNØR, IVØPT, NIR, NIZ, I1, I0)



2.2.9 Subroutine JØMIN (NADD, I1, I0)*

The input of the geometry data is controlled by the JØMIN subroutine, which performs the following tasks:

- Reads all geometry input data except the region volumes.
- Writes the body and zone data on a mass storage unit (IØUT=16).
- Determines the length of all geometry arrays.
- Calculates the beginning location in blank common of geometry arrays.
- Initializes geometry arrays.
- Calls the GTVLIN subroutine which returns region volumes.

Since combinatorial geometry input data is dynamically allocated to conserve storage area, it is stored temporarily on a mass storage device. This allows the core storage requirements to be determined. Hence, much of the coding in JØMIN is similar to GENI, which reads the data on the mass storage device and puts it into blank common.

Called from: INPUT1

Subroutines called: GENI, GTVLIN

Commons required: Blank, GØMLØC, PAREM

Variables required:

All variables in GØMLØC, I1, I0

Variables input:

IVØPT, IDBG, NAZ, MRIZ(I), MMIZ(I)

Variables changed:

All variables in GØMLØC, IVØPT, IDBG, MRIZ(I), MMIZ(I).

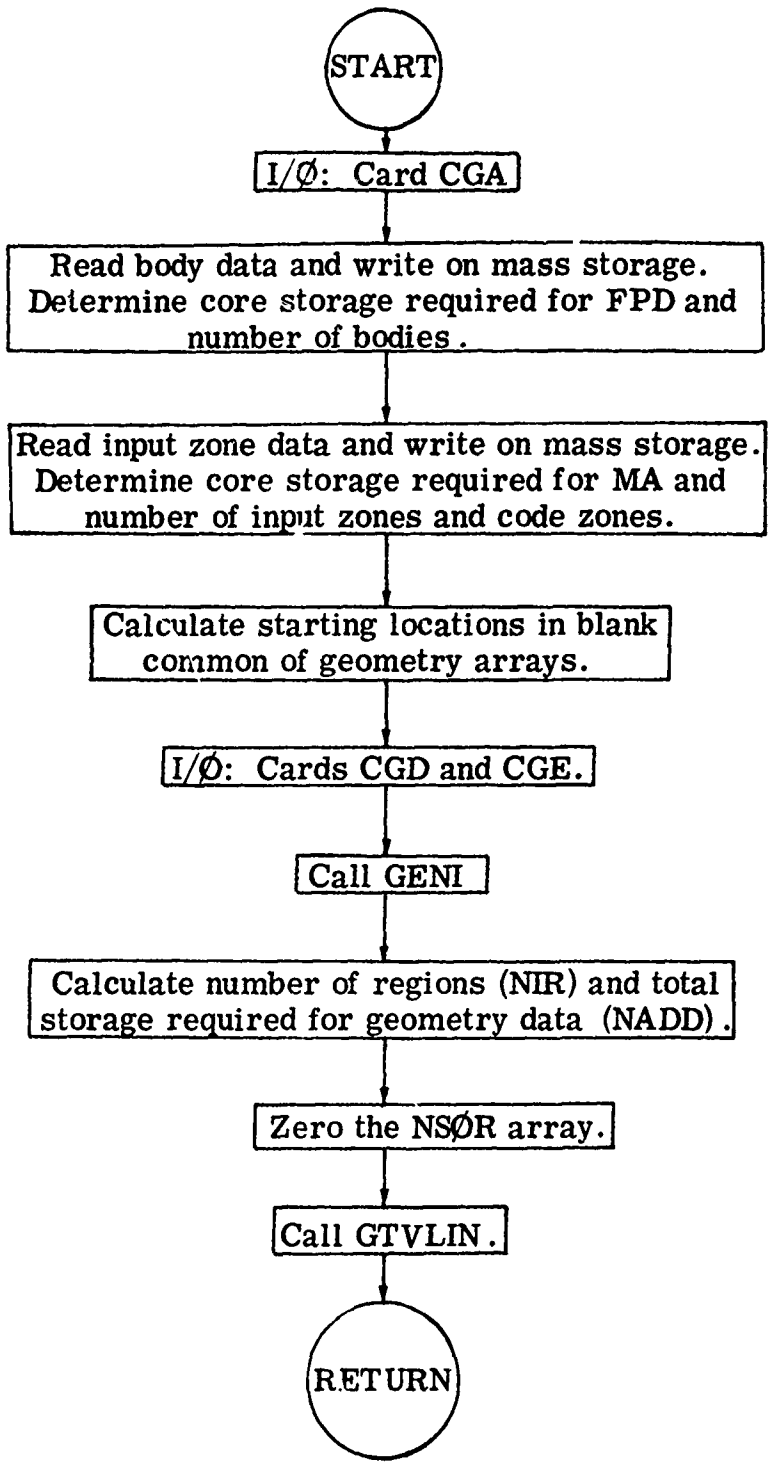
---

* CG version.

Important internal variables:

IØUT	-	mass storage unit,
NAZT	-	total number of adjacent zones summed over all zones.

Subroutine JØMIN (NADD, I1, I0)



2.2.10 Subroutine GTVØL (MXREG, GNØR)

This routine is called by ENDRUN to calculate the reciprocal of the volume of each region. It is geometry independent, since it uses information determined in GTVLIN.

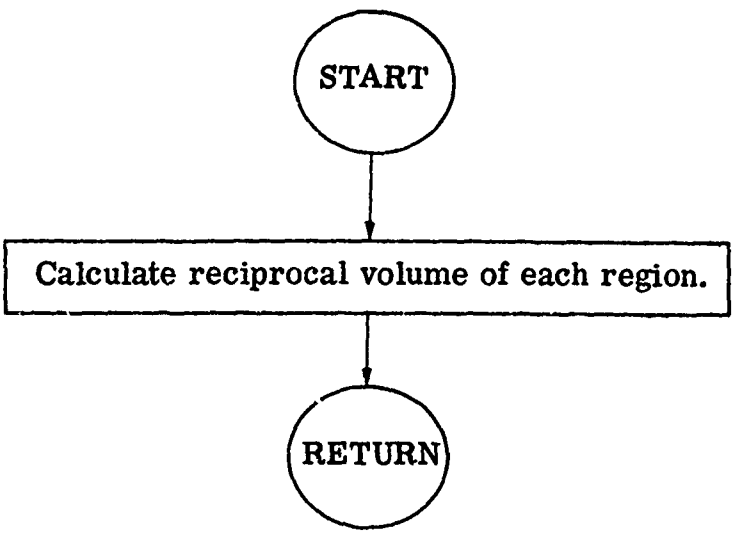
Called from: ENDRUN
Subroutines called: none
Commons required: GØMLØC, blank
Variables required:

KVØL - index from common GØMLØC,
MXREG - number of regions in the geometry.

Variables changed:

GNØR - reciprocal of the volume for each region.

Subroutine GTVØL (MXREG, GNØR)



2.2.11 Subroutine GØMST (TSIG, MARK)*

Any boundary crossing between the present and next collision sites are determined by calling the combinatorial geometry routine G1. Before the G1 call, combinatorial geometry variables in common PAREM are initialized, and after the call NUTRØN variables are updated. WATE*S is summed in blank common as a track length flux estimator. MARK is set to -1 for an external void and -2 for an internal void.

Called from: NXTCØL

Subroutines called: G1

Commons required: NUTRØN, APØLLØ, PAREM, ØRGI, GØMLØC

Variables required:

X, Y, Z, U, V, W, NMED see common NUTRØN for definitions.

BLZNT	-	value of IR from last track or from LØØKZ,
XB(3)	-	starting coordinates of present trajectory,
WB(3)	-	trajectory direction cosines,
MARK	-	flag to indicate type of trajectory,
ETATH	-	distance to be travelled in cm if the flight remains in the same media,
DIST	-	present distance from XB(3).

Variables changed:

X, Y, Z	-	endpoints of flight,
DIST0	-	distance from XB(3) to next collision site,
ETAUSD	-	mean free paths travelled on this call to GØMST,
ETATH	-	cm travelled on this call to GØMST,

* CG version.

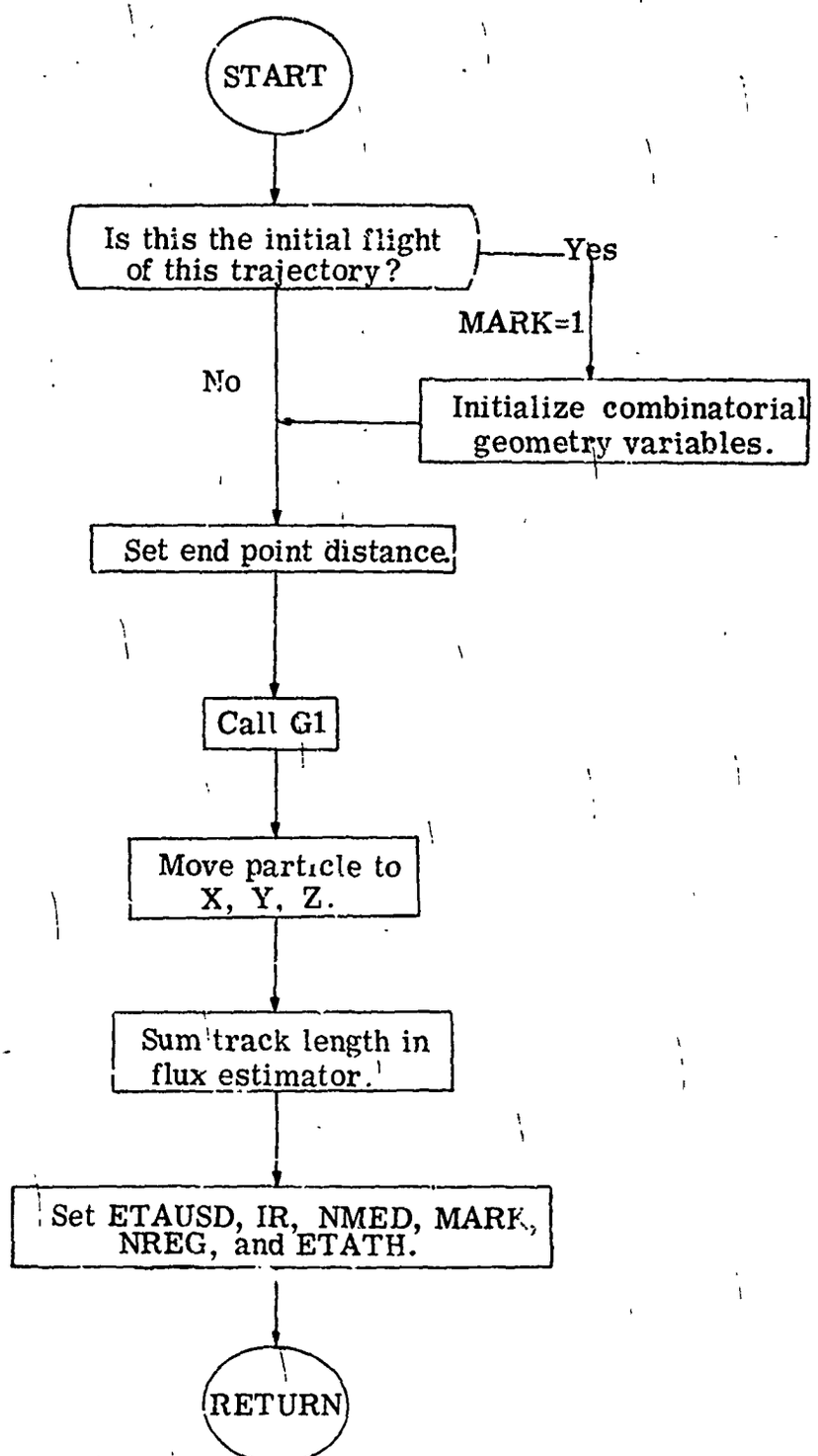
MARK - flag indicating type of termination of flight,

- 0 - normal boundary crossing,
- 1 - flight within one medium,
- -1 - particle escaped,
- -2 - particle entered an interior void,

NMED and NREG - medium and region of end point,

BLZNT - combinatorial geometry region of end point.

Subroutine GØMST (TSIG, MARK)



2.2.12 Subroutine EUCLID (MRK, X1, Y1, Z1, X2, Y2, Z2, P1P2, IG, ARG, NT, MEDIUM, IBLZ, NREGN)*

Subroutine EUCLID is used to determine the number of mean free paths between two points. However, if the argument NT is non-zero only the mean free paths to the first media boundary will be calculated. The routine works much like NXTCØL and GØMST. After initializing the combinatorial geometry variables, the particle is tracked from point one to point two. Total cross sections are determined by calling NSIGTA and distances to next interfaces are calculated by calling G1. On an initial call to EUCLID, MRK should be 1. If NT is non-zero, MRK should be 0 on successive calls for the same trajectory, so that new trajectory parameters are not initialized. This will be handled automatically if the calling routine does not change MRK.

Called from: GETETA, RELCØL

Subroutines called: G1, NSIGTA

Commons required: PAREM, ØRGI

Variables required:

MRK	-	set to 1 for initial call of a trajectory,
X1, Y1, Z1	-	coordinates of starting point,
X2, Y2, Z2	-	coordinates of end point,
P1P2	-	distance between starting and end points,
IG	-	energy group index,
NT	-	0 for total mean free paths, $\neq 0$ for mean free paths between intersection points,
MEDIUM	-	NMED of point one,
NREGN	-	NREG of point one,
IBLZN	-	IR of point one.

* CG version.

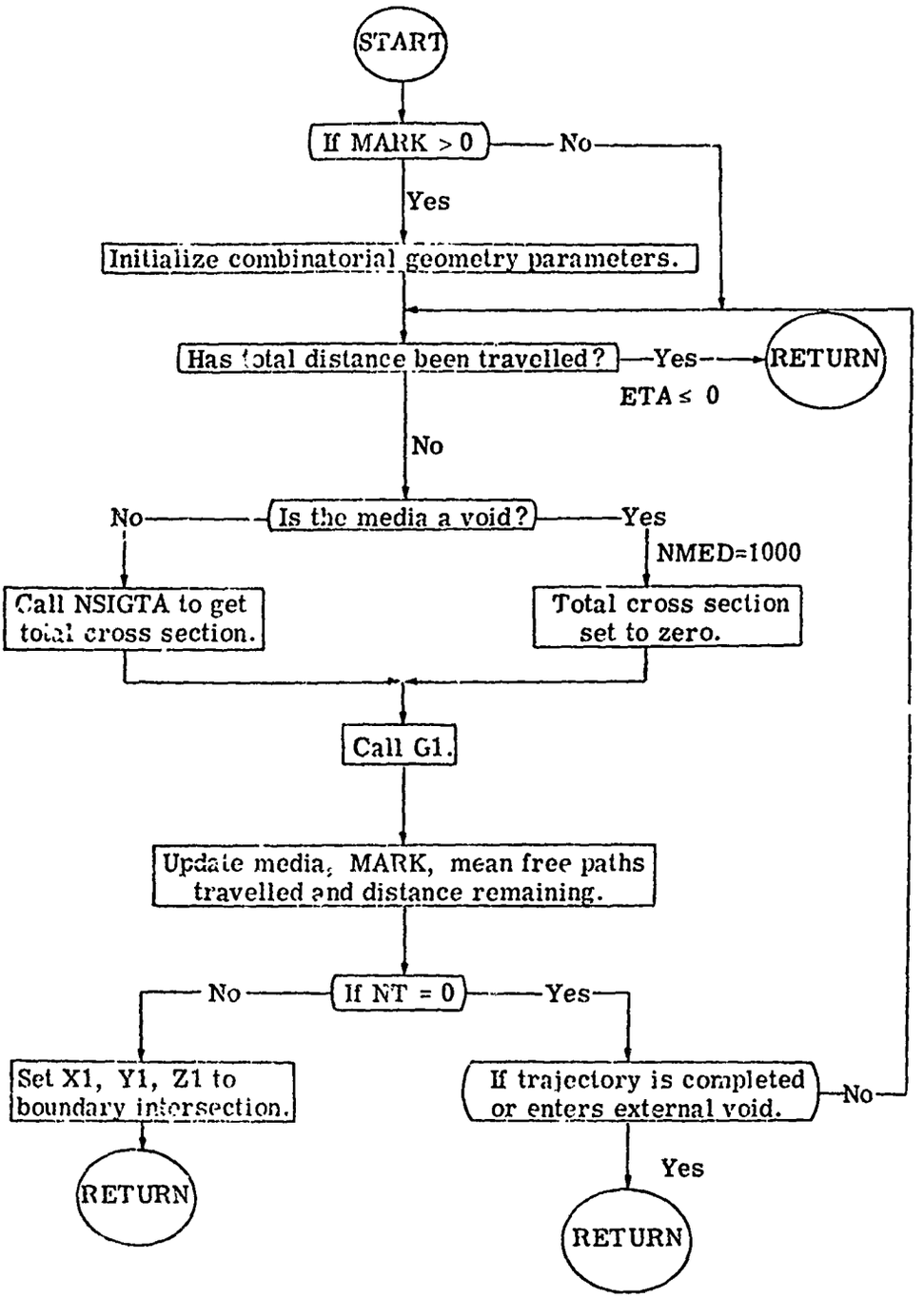
Variables changed:

MRK	-	1 for a flight reaching the end point,
	-	0 for a flight crossing a medium boundary (NT $\neq$ 0 only),
	-	-1 for a flight escaping the system,
	-	-2 for a flight encountering an internal void (NT $\neq$ 0 only),
X1, Y1, Z1	-	returns boundary intersection point if NT $\neq$ 0,
ARG	-	negative of number of mean free paths,
NT	-	if NT $\neq$ 0 on input, will return as -1 if an escape occurs,
MEDIUM	-	medium number of end point.

Significant internal variables:

ETA	-	distance remaining to point two in cm,
ETAUSD	-	distance travelled on last call of G1.

Subroutine EUCLID



2.2.13 Subroutine ENDRUN

This subroutine is called at the end of a run. The weight of the particles at collision and the track length of the particles has been summed over all collisions within the region for all particles in the run. Thus, a normalization by the number of particles in the run is required. In addition, the volume of the region and the total cross section for each region and energy group is used to determine the average fluence over the regions for the collision density estimate. For the track length estimate the same coding is used with the total cross section set to 1. Note that a region cannot contain more than one medium. Since storage allocations are flexibly dimensioned, NEX in the SAMBØ input must provide for MXREG + 2 arrays in Blank Common to be used by ENDRUN. A check is made and if this condition is not met, no output from ENDRUN is obtained.

There are many manipulations in ENDRUN with the energy indexes. These manipulations are required in order to treat the six major options in MØRSE. Appendix A gives an example of the six problems.

Called from: NRUN

Subroutines called: GTVØL, NSIGTA

Commons required: blank, PDET, USER

Variables required:

NMED (REGIØN) - must be set in data statement,

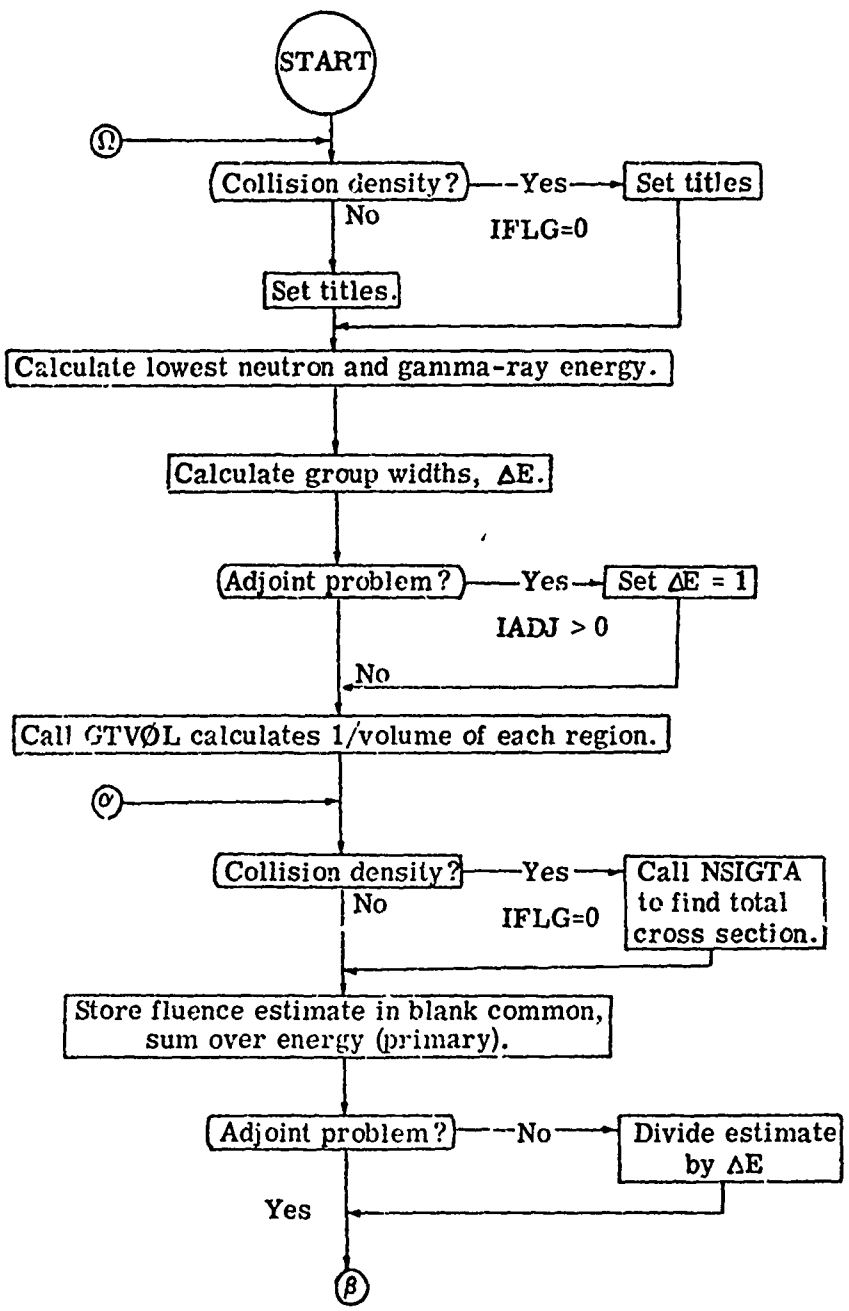
GNØR - reciprocal of the volume for each region,

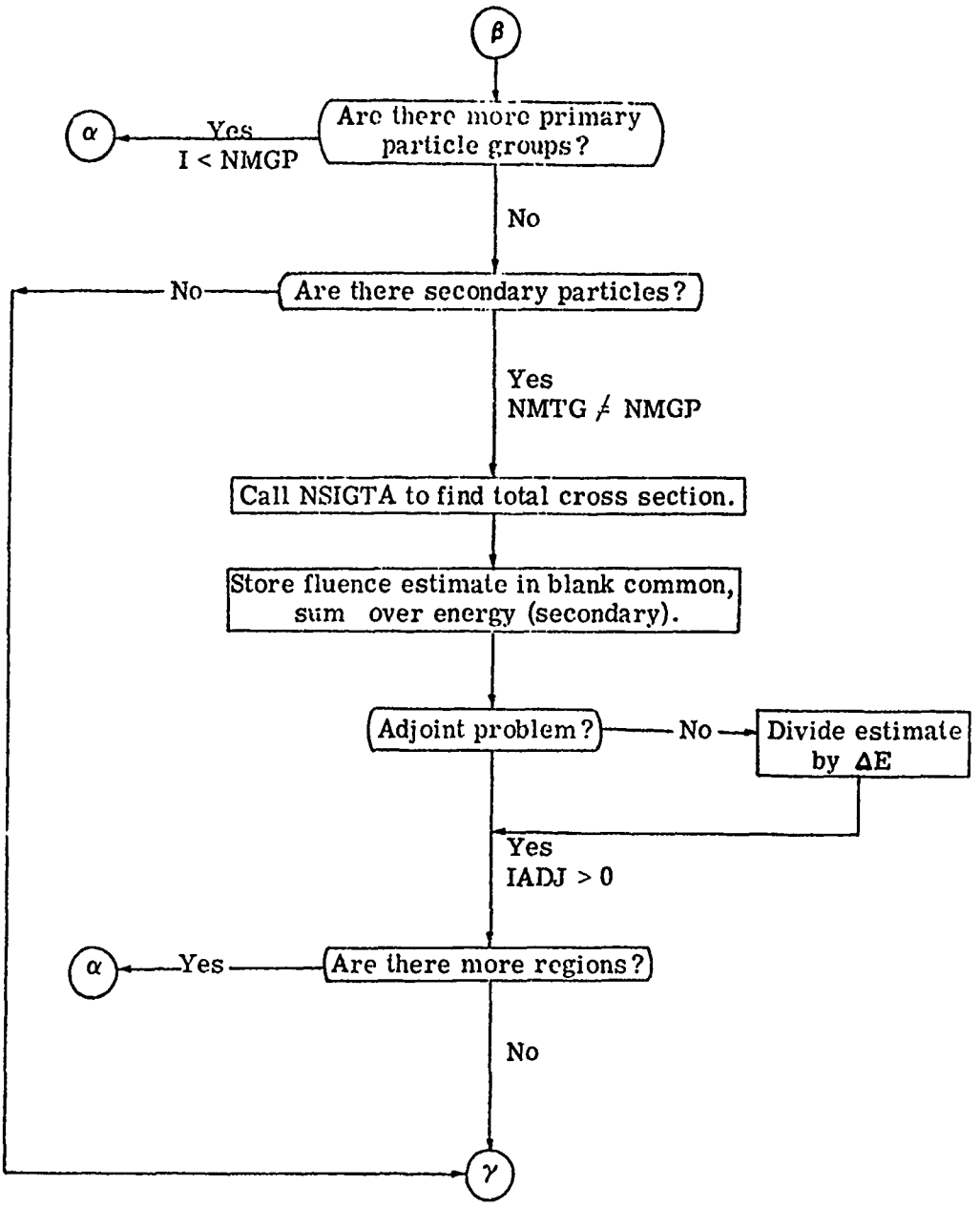
Many variables in blank common.

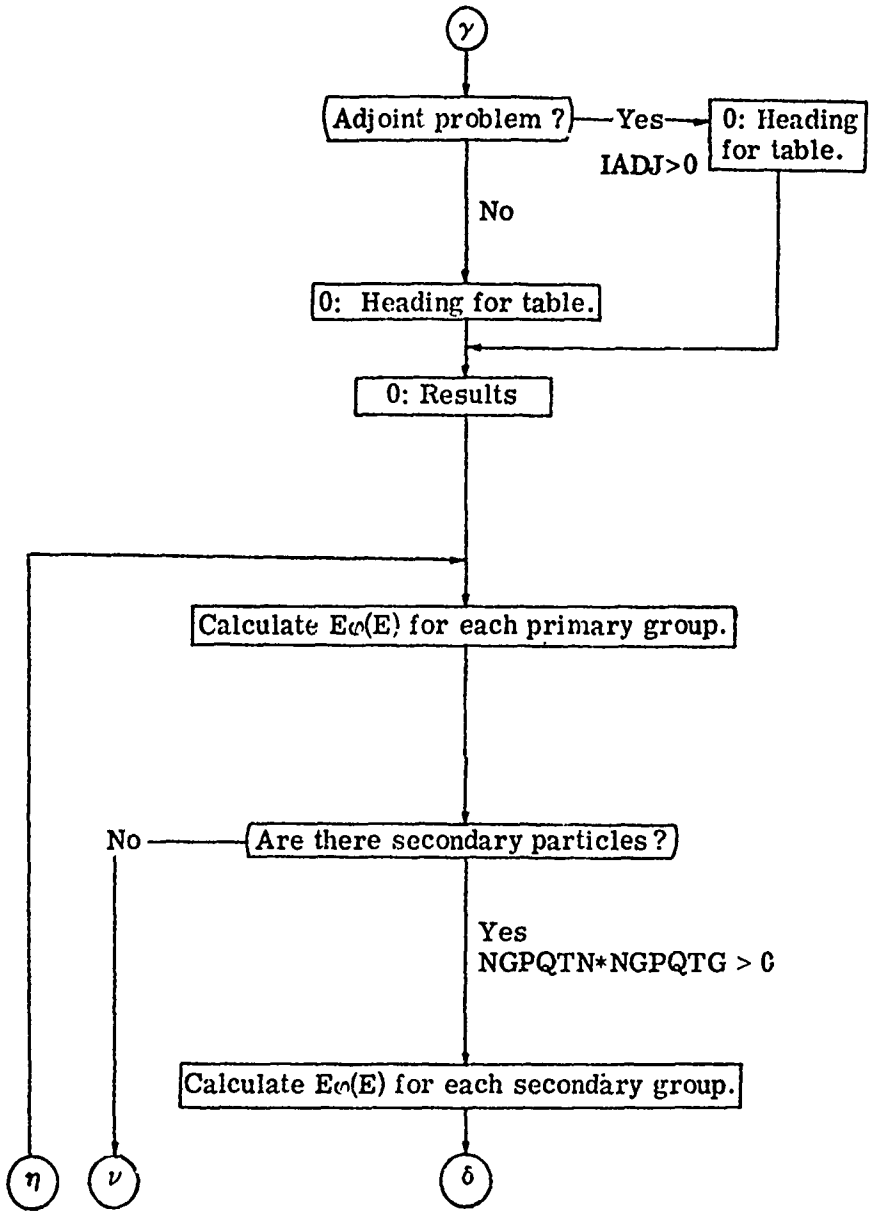
Significant internal variables:

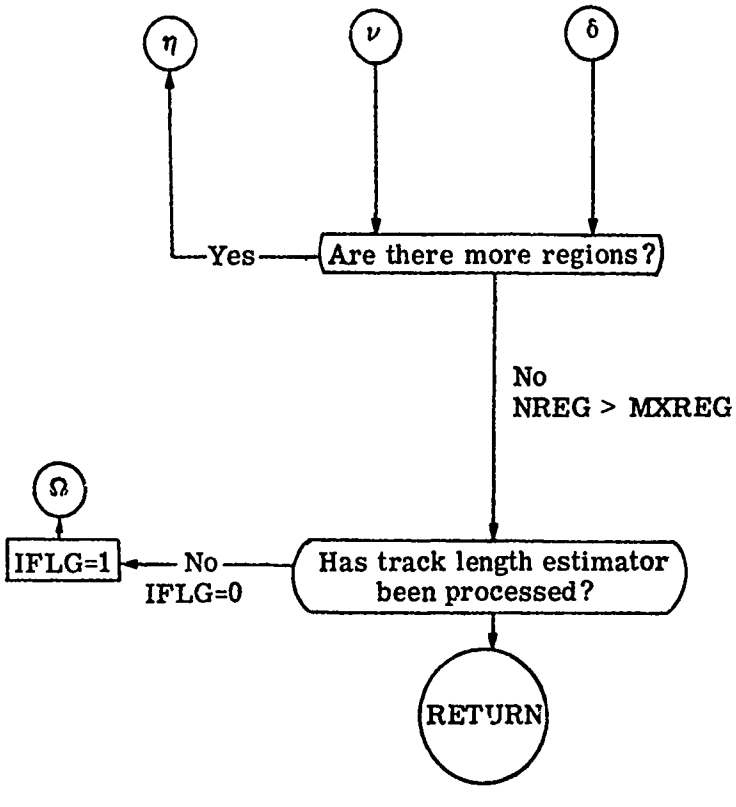
FNT	-	total number of source particles in the run,
MXREG	-	the number of geometry regions,
LIM1	-	the group number for the last primary particle group,
EBØT1	-	lowest primary particle group energy,
EBØT2	-	lowest secondary particle group energy,
NGP3	-	the group number for last secondary group,
NQP1	-	the group number for last primary group (negative if gamma-ray only problem),
IFLG	-	flag to indicate which fluence estimate is currently being output. 0-collision density, 1-track length.

Subroutine ENDRUN




β
Are there more primary particle groups?
Yes
I < NMGP
α
No
Are there secondary particles?
No
Yes
NMTG ≠ NMGP
Call NSIGTA to find total cross section.
Store fluence estimate in blank common, sum over energy (secondary).
Adjoint problem?
No
Divide estimate by ΔE
Yes
IADJ > 0
Are there more regions?
Yes
α
No
γ


γ
Adjoint problem ?
Yes
IADJ>0
0: Heading for table.
No
0: Heading for table.
0: Results
Calculate Eω(E) for each primary group.
No
Are there secondary particles ?
Yes
NGPQTN*NGPQTG > 0
Calculate Eω(E) for each secondary group.
η
ν
δ


η
ν
δ
Yes
Are there more regions?
No
NREG > MXREG
Ω
IFLG=1
No
IFLG=0
Has track length estimator been processed?
RETURN

III. MODIFICATIONS FOR UNIVAC-1108

In converting the MØRSE-SAMBØ-MØRSEC code from the IBM 360 to the UNIVAC-1108, there were a considerable number of changes that had to be made. These changes were connected mostly with the local library functions of the IBM-360 and an attempt was made to not add local UNIVAC-1108 library functions. The only reduction in the code's capability is in the diagnostic module where the ability to look at a bit string and decide whether it was a floating point or integer number, convert it to hollerith and output it, does not exist in this version. However, the diagnostic module is mostly machine independent and still has the capability of writing out values from common and parts of blank common. Subroutines BNKHLP and HELPER are dummy routines. A diagnostic routine for the combinatorial geometry package has been added.

Routines which are still machine dependent (involving coding other than word lengths, data statements and equivalences) are DATE and TIMER. Subroutine DATE calls ERTRAN and uses DECØDE. Subroutine TIMER calls CPUTIM that assumes that the time is returned in units of microseconds.

Subroutine READSG is still machine dependent in its need to read each card twice and subroutine ERRØR has been added--its only function is to call EXIT. Subroutine INPUT now calls INPUT1 and INPUT2 with this change, permitting a better use of overlay. The first six variables in NXTRA from common APØLLØ are used.

The layout of blank common is given in Figs. 12-18 for the different modules of MØRSE. Definitions of variables in blank common are given in Table III which defines the indexing scheme. Variables in the labelled commons are defined in Tables IV - XVI.

Preceding page blank

A description of the energy indexing scheme in MØRSE-SAMBØ-MØRSEC is given in Appendix A.

3.1 Built-In Fluence Estimators

There are two fluence estimators that are "built-in" the UNIVAC 1108 combinatorial geometry version of MØRSE. The addition of the collision density estimator to MØRSE has been discussed previously[7] and the track length per unit volume estimator has been added to take advantage of a feature of the combinatorial geometry. The resulting fluence is averaged over geometry regions and if no regions are used only the spectrum averaged over the whole system is available.

3.2 Collision Density Fluence Estimator

An estimate of the fluence averaged over a region specified by the geometry input may be obtained from the weight that has been summed over all collisions in the region by

$$\varphi_V(E_i) = \frac{\sum_{n=1}^{N_{Vi}} w_n}{NSTRT*NITS*VOL_V*\Sigma_{TV}(E_i)} \quad ,$$

where

$\varphi_V(E_i)$ = fluence in region V at energy E_i,

N_{Vi} = number of particles of energy E_i scattering in region V,

w_n = weight of the nth particle of energy E_i scattering in region V,

NITS*NSTRT = total number of particles,

VOL_V = volume of region V,

$\Sigma_{TV}(E_i)$ = total cross section for groups E_i in region V.

3.3 Track Length per Unit Volume Estimator

Another estimate of the fluence averaged over a region may be obtained from the track length of the neutron or photon in that region. The track length calculated by the combinatorial geometry package is stored in a manner similar to the collision weights. The fluence estimate may be obtained by

$$\varphi_V(E_i) = \frac{\sum_{n=1}^{N_{Vi}} L_n \, w_n}{NSTRT*NITS*VOL_V} \quad ,$$

where

N_{Vi} = number of trajectories of energy E_i in region V,

w_n = weight of particle on n^{th} trajectory, and

L_n = track length of nth trajectory in region V at energy E_i.

Subroutine ENDRUN is called to process the weights and track lengths to obtain estimates of the fluence. In an adjoint problem, the importance is calculated. The fluence estimate has units of particles/eV/cm^2/source and in an adjoint problem the units are importance/cm^2/source. The total fluence is integrated over energy from the primary and secondary energy groups to obtain an integral value for each region. The only input required for this routine is a data statement giving the medium number for each region; there is a limitation of only one medium per region. In addition, one of the several options for inputing or calculating the volume of each region must be utilized.

The major disadvantage with using these estimators is that no estimate of statistics is available. The use of this estimator is inexpensive, however (output routine is called only once), and is meant to be used as a supplemental estimator.

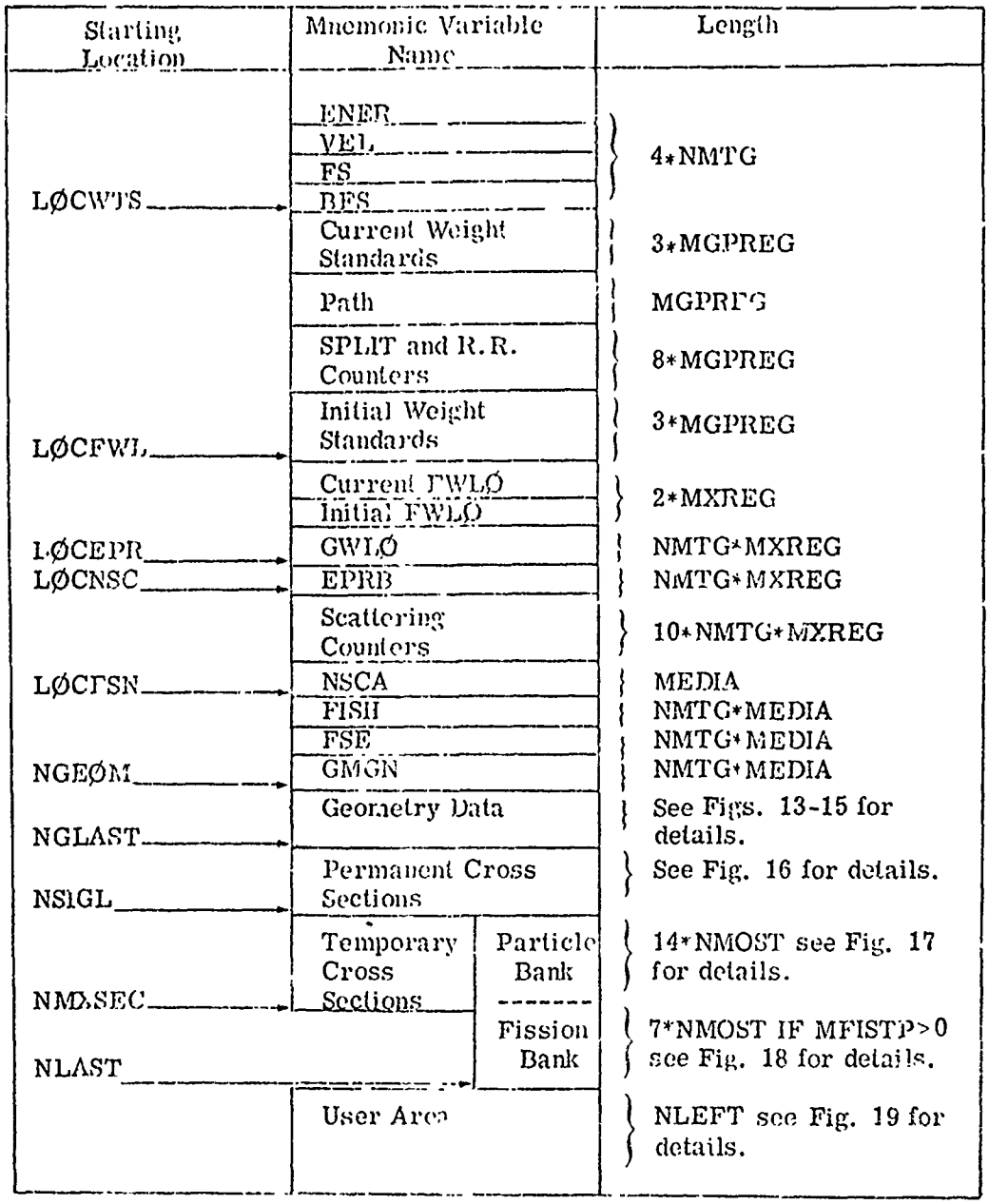


Fig. 12. General layout of blank common.

Starting Location	Information	Size
NGEØM=NADD	Length of geometry array	1
KMA	MA Integer array	UTMA
KFPD	FPD Floating point array	LFPD
KLCR	LØCREG Indices to correlate MA array data with code zone data	NUMR
KNBD	NUMBØD Number of bodies for each code zone	NUMR
KIØR	IRØR Indices to correlate input zone to code zone	NUMR
KRIZ	MRIZ Indices to correlate MØRSE region to input zone	IRTRV
KRCZ	MRCZ Indices to correlate MØRSE region to code zone	NUMR
KMIZ	NMIZ Indices to correlate MØRSE media to input zone	IRTRV
KMCZ	NMCZ Indices to correlate MØRSE media to code zone	NUMR
KKR1	KR1 Indices to correlate first code zone to input zones	IRTRV
KKR2	KR2 Indices to correlate last code zone to input zone	IRTRV
KNSR	NSØR Indices of code zones in which source particles have been found	NUMR
KVØL	VNØR Volume of each MØRSE region	NIR
NGLAST		

Fig. 13. Layout of combinatorial geometry data in blank common.

Position in Blank Common	Information Stored	Size	Description
KFPD	RIN for body 1	1	Path length data for last trajectory in body 1.
KFPD + 1	RØUT for body 1	1	
KFPD + 2	First 6 words of real data for body 1	6	Read from first card of body 1 card set.
KFPD + 8	Remaining words of real data for body 1	N_1	N_1 depends on body type. See Table B IV.
KFPD + 8 + N_1	RIN for body 2	1	Same information as above, but for body 2.
	RØUT for body 2	1	
	Remaining data for body 2 (N_2 words)	N_2	
	Repeat for NUMB bodies.		

NOTE: 8 words are set aside at the end of the FPD array, but are not used.

Fig. 14. Detailed layout of the FPD array in blank common.

Position in Blank Common	Information Stored	Size	Description
KMA	Body number 1	7	Body number 1 data (each body requires 7 words of information).
KMA + 1	LØØP for body number 1		
KMA + 2	Body type (ITYPE)		
KMA + 3	LRI		
KMA + 4	LRØ		
KMA + 5			
KMA + 6	Beginning location in FPD of body data		
KMA + 7	Body number 2	7	Same information and order as for body number 1.
KMA + 14	↓		
KMA (L-1)*7	Body number (L)	7	Body number L is the last body.

Figure 15.

Position in Blank Common	Information Stored	Size	Description
KMA + L*7	Zone number 1	1	Beginning of code zone information.
	Number of first body in this zone Location of integer data for this body First zone to search upon exiting this body Location of next zone to be searched	4	Beginning of information about bodies defining code zone 1. Integer data location is given by: 7*(Body number) - 6.
	Data on second body in this zone	4	The last two words in each set of body data initiate the "leap frog" process by which the code stores possible zones which can be entered upon exiting this body in that particular zone. These zones are checked by the code when the next zone entered is being determined. If the next zone is not located from this stored data, all zones are searched.
	Data on last body in the zone	4	

Figure 15.

Position in Blank Common	Information Stored	Size	Description
	Zone number 2	1	
	Body information		Same information as above except for code zone number 2.
	Zone data of last zone		Code zone information about the last zone input on cards.
KMA + LDATA	Code search information KMA + LTMA-1	2*NAZT	Storage set aside for determining the zone to be searched and where the next zone number is located.

Fig. 15. Detailed layout of the MA array in blank common.

Location	Information	Size
IRSG = NLAST = NGLAST+1	List of Mixing Table	3*NMIX
INGS	Index to Σ_{gg} (Primary)	NGP
INSG	Index to Σ_{gg} (Secondary)	NGG
IDEL	List of Element I.D. Numbers	NELEM*NCØEF if IXTAPE > 0
INNN	Number of Downscatters for each primary group	NGP
IGGG	Number of Downscatters for each secondary group	NGG
ISTART	Σ_T	NTG
ISCCØG	Σ_s	NTG
INABØG	Σ_s/Σ_T	NTG
IGABØG	Σ_γ/Σ_T	NGP
IFPØRG	$\nu\Sigma_f/\Sigma_T$	NTG
INUS	$\Sigma_{g'\to g}$	NTG
INGN	Σ_N/Σ_T not used at present	NGG
INGNP	$\Sigma_{\gamma\to N}$ not used at present	NGG*NGP
IFNGP	$\Sigma_{N\to\gamma}$	NGP*NGG
IFSPØG	$\Sigma^N_{g'\to g}$	NDSNGP
IDSGØG	$\Sigma^\gamma_{g'\to g}$	NDSNGG
IPRBNG	$P^N_{g'\to g}$	NDSNGP*NSCT
IPRBGG	$P^\gamma_{g'\to g}$	NDSNGG*NSCT
ISCANG	$A^N_{g'\to g}$	NDSNGP*NSCT
ISCAGG	$A^\gamma_{g'\to g}$	NDSNGG*NSCT
ISPØRG + ISTART If ISTAT > 0 ISPØRT	Repeat for next medium	ISPØRG
	P_1 Coefficient Primary	NDSNGP
	P_1 Coefficient Secondary	NDSNGG
	Repeat for P_L Coefficient	INFPØG*(NCØEF-2)
NLAST or NNIC	Repeat for next medium	INFPØG*(NCØEF-1)
NNIC	Index to point cross sections for each multigroup boundary	(IGQPT+1)*NMED if IØ6RT>0; otherwise zero

Location	Information		Size
NXSECT(1)	Point cross sections for Medium 1	total	NPT(1)*(NEGPS-1)*NXPM
		scattering	
		$\nu\Sigma_f$	
NXSECT(J)	Point cross sections for Medium J	total	NPT(J)*(NEGPS-1)*NXPM
		scattering	
		$\nu\Sigma_f$	
NLAST			

Fig. 16. Layout of permanent cross sections in blank common.

Location	Variable	
NNØ = NSIGL	IG	
	U	
	V	
	W	
	X	
	Y	Variables are from NUTRØN common, see Table VI for definitions.
	Z	
	WATE	
	AGE	
	BLZNT	
	NAME	
	NAMEX	
	NMED	
	NREG	
NNØ + 14		
	Repeat for Particle 2	
	.	
	.	
NNØ + 14*NMØST = NLAST		
= NFISBN IF MFISTP > 0		

Fig. 17. Layout of particle bank in blank common.

Location	Variable
NFISBN	X
	Y
	Z
	WATEF
	AGE
	IG
	NAMEX
NFISBN + 7	
	Repeat for particle 2
	.
	.
	.
NFISBN + 7*NMØST = NLAST	

Fig. 18. Layout of fission bank in blank common.

Location Labels	Mnemonic Variable Name	Length
NLAST* + 1		
LØCRSP	LABELS	20*NRESP
	RESP	NRESP*NMTG
LØCXD	EXTR	NEX*NMTG
	XD YD ZD RAD TØ FACT	6*ND
LØCIB	EXTR	NEXND*ND
	IB EP DELE	3*NE
LØCCØ		
LØCT	CØS	NA
	T DELT	2*ND*NT
LØCUD		
	UD SUD SUD2	3*ND*NRESP
LØCSD		
	SD SSD SSD2	3*ND*NRESP
LØCQE	SUD & SSD Units	20

*NOTE: NLAST must not be changed after SCØRIN is called.

Location Labels	Mnemonic Variable Name	Length
	QE SQE SQE2	3*NE*ND
LØCQT →	SQE Units	20
	QT SQT SQT2	3*NT*ND*NRESP
LØCQTE →	SQT Units	20
	QTE SQTE SQTE2	3*NT*NE*ND
LØCQAE →	SQTE Units	20
	QAE SQAE SQAE2	3*NA*NE*ND
LMAX →	SQAE Units	20

Fig. 19. Layout of user area (analysis) in blank common.

TABLE III.

Indexing of Random Walk Blank Common Arrays

Mnemonic Variable Name	Location of Array in Blank Common [BC(I) or NC(I)]
ENER(IG)	BC(I) ; I = IG
VEL(IG)	I = NMTG + IG
FS(IG)	I = 2*NMTG + IG
BFS(IG)	I = 3*NMTG + IG
WTHI(IG, NREG)	BC(I) ; I = LØCWTS + (NREG-1)*MAXGP + IG
WTLØ(IG, NREG)	I = LØCWTS + MGPREG + (NREG-1)*MAXGP + IG
WTAV(IG, NREG)	I = LØCWTS + 2*MGPREG + (NREG-1)*MAXGP + IG
PATH(IG, NREG)	I = LØCWTS + 3*MGPREG + (NREG-1)*MAXGP + IG
NSPL(IG, NREG)	NC(I) ; I = LØCWTS + 4*MGPREG + (NREG-1)*MAXGP + IG
WSPL(IG, NREG)	I = LØCWTS + 5*MGPREG + (NREG-1)*MAXGP + IG
NØSP(IG, NREG)	I = LØCWTS + 6*MGPREG + (NREG-1)*MAXGP + IG
WNØS(IG, NREG)	I = LØCWTS + 7*MGPREG + (NREG-1)*MAXGP + IG
RRKL(IG, NREG)	I = LØCWTS + 8*MGPREG + (NREG-1)*MAXGP + IG
WRKL(IG, NREG)	I = LØCWTS + 9*MGPREG + (NREG-1)*MAXGP + IG
RRSU(IG, NREG)	I = LØCWTS + 10*MGPREG + (NREG-1)*MAXGP + IG
WRSU(IG, NREG)	I = LØCWTS + 11*MGPREG + (NREG-1)*MAXGP + IG

TABLE III. (Cont'd.)

Mnemonic Variable Name	Location of Array in Blank Common [BC(I) or NC(I)]
INIWHI(IG, NREG)	BC(I) ; I = LØCWTS + 12*MGPREG + (NREG-1)*MAXGP + IG
INIWLØ(IG, NREG)	I = LØCWTS + 13*MGPREG + (NREG-1)*MAXGP + IG
INIWAV(IG, NREG)	I = LØCWTS + 14*MGPREG + (NREG-1)*MAXGP + IG
FWLØ(NREG)	BC(I) ; I = LØCFWL + NREG
INIFLØ(NREG)	I = LØCFWL + MXPTG + NREG
GWLØ(IG, NREG)	I = LØCFWL + 2*MXREG + (NREG-1)*NMTG + IG
EPRB(IG, NREG)	BC(I) ; I = LØCEPR + (NREG-1)*NMTG + IG
NSCT(IG, NREG)	NC(I) ; I = LØCNSC + (NREG-1)*NMTG + IG
WSCT(IG, NREG)	BC(I) ; I = LØCNSC + NMTG*MXREG + (NREG-1)*NMTG + IG
NALB(IG, NREG)	NC(I) ; I = LØCNSC + 2*NMTG*MXREG + (NREG-1)*NMTG + IG
WALB(IG, NREG)	BC(I) ; I = LØCNSC + 3*NMTG*MXREG + (NREG-1)*NMTG + IG
NFIZ	NC(I) ; I = LØCNSC + 4*NMTG*MXREG + (NREG-1)*NMTG + IG
WFIZ	BC(I) ; I = LØCNSC + 5*NMTG*MXREG + (NREG-1)*NMTG + IG

TABLE III. (Cont'd.)

Mnemonic Variable Name	Location of Array in Blank Common [BC(I) or NC(I)]
NGAM (IG)	NC(I) ; I = LØCNSC + 6*NMTG*MXREG + (NREG-1)*NMTG + IG
WGAM (IG)	BC(I) ; I = LØCNSC + 7*NMTG*MXREG + (NREG-1)*NMTG + IG
NTRK (IG)	NC(I) ; I = LØCNSC + 8*NMTG*MXREG + (NREG-1)*NMTG + IG
WTRK (IG)	BC(I) ; I = LØCNSC + 9*NMTG*MXREG + (NREG-1)*NMTG + IG
NSCA(IMED)	BC(I) ; I = LØCNSC + IMED + 10*NMTG*MXREG
	or I = LØCFSN - MEDIA + IMED
FISH(IG, IMED)	BC(I) ; I = LØCFSN + (IMED-1)*NMTG + IG
FSE(IG, IMED)	I = LØCFSN + NMTG*MEDIA + (IMED-1)*NMTG + IG
GMGN(IG, IMED)	I = LØCFSN + 2*NMTG*MEDIA + (IMED-1)*NMTG + IG

TABLE IV.

Indexing of Analysis Arrays in Blank Common

Mnemonic Variable Name	Location in Blank Common [BC(I) or NC(I)]	Description
LABEL(J, NR)	I = NLAST + (NR-1)*20 + J J = 1, 20	80-character Hollerith label for each response function.
RESP(IG, NR)	I = LØCRSP + (NR-1)*NMTG + IG	Value of response function.
EXTR(IG, NX)	I = LØCRSP + NRESP*NMTG + (NX-1)*NMTG + IG	Extra arrays of length NMTG.
XD(ID)	I = LØCXD + ID	x-coordinate of detector ID.
YD(ID)	I = LØCXD + ND + ID	y-coordinate of detector ID.
ZD(ID)	I = LØCXD + 2*ND + ID	z-coordinate of detector ID.
RAD(ID)	I = LØCXD + 3*ND + ID	Distance from source (assumed to be at XSTRT, YSTRT, ZSTRT) to detector.
TØ(ID)	I = LØCXD + 4*ND + ID	Minimum flight time to detector.
FACT(ID)	I = LØCXD + 5*ND + ID	Detector-dependent normalization
EXTR(ID, NXN)	I = LØCXD + 6*ND + (NXN-1)*ND + ID	Extra arrays of length ND.
IB(IE)	I = LØCIB + IE	Energy bin group number.

TABLE IV. (Cont'd.)

Mnemonic Variable Name	Location of Blank Common [BC(I) or NC(I)]	Description
EP(IE)	I = LØCIB + NE + IE	Lower energy of bin.
DELE(IE)	I = LØCIB + NE + IE	Widths of bin.
CØS(IA)	I = LØCCØ + IA	Cosine of angle bin limits.
T(IT, ID)	I = LØCT + (ID-1)*NT + IT	Time bin limits.
DELT(IT, ID)	I = LØCT + ND·NT + (ID-1)*NT + IT	Widths of time bins.
UD(NR, ID)	I = LØCUD + (ID-1)*NRESP + NR	Uncollided response.
SD(NR, ID)	I = LØCSD + (ID-1)*NRESP + NR	Total response.
QE(IE, ID)	I = LØCQE + (ID-1)*NE + IE	Energy-dependent fluence.
QT(NR, IT, ID)	I = LØCQT + (ID-1)*NTNR + (IT-1)*NRESP + NR	Time-dependent response.
QTE(IT, IE, ID)	I = LØCQTE + (ID-1)*NTNE + (IE-1)*NT + IT	Time and energy-dependent fluence.
QAE(IA, IE, ID	I = LØCQAE + (ID-1)*NANE + (IE-1)*NA + IA	Angle and energy-dependent fluence.

Index	Maximum Value	Definition
NR	NRESP	Response function.
IG	NMTG	Energy group.

TABLE IV. (Cont'd.)

Index	Maximum Value	Definition
NX	NEX	Extra array (length NMTG).
ID	ND	Detector.
NXN	NEXND	Extra array (length ND).
IE	NE	Energy bin (one or more groups).
IA	NA	Angle bin.
IT	NT	Time bin.

TABLE V.

Definition of Variables in APØLLØ Common

Variable	Definition
AGSTRT	Input starting age of source particle.
DDF	Starting particle weight as determined in SØRIN.
DEADWT(5)	The summed weights of the particles at death. The four deaths are: Russian roulette, escape, energy, and age limit. DEADWT(5) is unused.
ETA	Mean-free-path between collisions.
ETATH	Distance in cm to the next collision if the particle does not encounter a change in total cross section.
ETAUSD	Flight path in m.f.p. that has been used since the last event.
UINP, VINP, WINP	Input direction cosines for source particle.
WTSTRT	Input starting weight.
XSTRT, YSTRT, ZSTRT	Input starting coordinates for source particle.
TCUT	Age limit at which particles are retired.
XTRA(10)	Not used.
I0, I1	Output and input logical units.
MEDIA	Number of media for which there are cross sections.
IADJM	Switch indicating an adjoint problem if > 0.
ISBIAS	Switch indicating that source energy distribution is to be biased if > 0.
ISØUR	Input source energy group if > 0; otherwise, SØRIN is called to read input spectrum.
ITERS	Number of batches still to be processed in the run.
ITIME	Not used.
ITSTR	Switch indicating that secondary fissions are to be the source for the next batch if > 0.

TABLE V. (Cont'd.)

Variable	Definition
LØCWTS	Starting location in blank common of the weight standards and other arrays MGPREG long.
LØCFWL	Starting location in blank common of the fission weights.
LØCEPR	Starting location in blank common of the energy-biasing parameters.
LØCNSC	Starting location in blank common of the scattering counters.
LØCFSN	Starting location in blank common of the fission and gamma-generation probabilities for each medium and group.
MAXGP	Maximum number of energy groups for which there are weight standards or path-length stretching parameters.
MAXTIM	The elapsed clock time at which the problem is terminated.
MEDALB	Medium number for the albedo medium.
MGPREG	Product of number of weight standard groups (MAXGP) and regions (MXREG).
MXREG	Maximum number of regions in the system.
NALB	An index indicating that an albedo scattering has occurred if > 0.
NDEAD(5)	Number of deaths of each type (see DEADWT).
NEWNM	Name of the last particle in the bank.
NGEØM	Location of first cell of geometry data storage in blank common
NGPQT1*	The lowest energy group (largest group number) for which primary particles are to be followed.
NGPQT2*	The number of primary particle groups.
NGPQT3*	The lowest energy group (largest group number) for which any particle is to be followed.

* See Appendix A for diagram of energy group structure.

TABLE V. (Cont'd.)

Variable	Definition
NGPQTG	Number of energy groups of secondary particles to be followed.
NGPQTN*	Number of energy groups of primary particles to be followed.
NITS	Number of batches per run.
NKCALC	The first batch to be used for a k calculation. If 0, k is not calculated.
NKILL	An index to indicate that Russian roulette is to be played if > 0.
NLAST	The last cell in blank common that was used by the cross-section storage or is set aside for banking.
NMEM	The location of the next particle in the bank to be processed.
NMGP*	The number of primary particle groups for which there are cross sections.
NMØST	The maximum number of particles that the bank can hold.
NMTG*	The total number of energy groups (both primary and secondary) for which there are cross sections.
NØLEAK	An index which indicates that nonleakage path-length selection is to be used if > 0.
NØRMF	An index to indicate that the fission parameters are to be renormalized if > 0.
NPAST	An index to indicate that the exponential transform is to be used if > 0.
NPSCL(13)	An array of counters of events for each batch: 1. sources generated 2. splittings occurring 3. fissions occurring 4. gamma rays generated

* See Appendix A for diagram of energy group structure.

TABLE V. (Cont'd.)

Variable	Definition
	5. real collisions
	6. albedo scatterings
	7. boundary crossings
	8. escapes
	9. energy cutoffs
	10. time cutoffs
	11. Russian roulette kills
	12. Russian roulette survivors
	13. gamma rays not generated because bank was full.
NQUIT	Number of runs still to be processed.
NSIGL	Starting location of the bank in blank common.
NSØUR	An index input to indicate that fissions are to be the source for future batches.
NSPLT	An index to indicate that splitting is to be considered if > 0.
NXTRA(1-5)	Used to separate INPUT1 and INPUT2.
NXTRA(6)	NGPREG calculated in MØRSE and used in GØMST as an index for track length estimation.
NXTRA(7-10)	Not used.

TABLE VI.

Definition of Variables in NUTRØN Common

Variable	Definition
NAME	Particle's first name.
NAMEX	Particle's family name. (Note that particles do not marry.)
IG	Current energy group index.
IGØ	Previous energy group index.
NMED	Medium number at current location.
MEDØLD	Medium number at previous location.
NREG	Region number at current location.
U, V, W	Current direction cosines.
UØLD, VØLD, WØLD	Previous direction cosines.
X, Y, Z	Current location.
WATE	Current weight.
ØLDWT	Weight at previous collision.
WTBC	Weight just before current collision.
BLZNT	Current block and zone number (packed) for Ø5R geometry, and current zone number IR for combinatorial geometry.
BLZØN	Previous block and zone number (packed) for Ø5R geometry, and previous zone number IR for combinatorial geometry.
AGE	Current age.
ØLDAGE	Previous age.

TABLE VII.

Definitions of Variables in USER Common

Variable	Definition
AGSTRT	Initial chronological age to be assigned to source particles.
WTSTRT	Initial weight to be assigned to source particles.
XSTRT	Initial x position to be assigned to source particles.
YSTRT	Initial y position to be assigned to source particles.
ZSTRT	Initial z position to be assigned to source particles.
DFF	Normalization for adjoint problems.
EBØTN	Lower energy boundary (eV) of last neutron group (group NMGP).
EBØTG	Lower energy boundary (eV) of last gamma-ray group (group NMTG).
TCUT	Chronological age limit.
I0	Logical unit for output.
I1	Logical unit for input.
IADJM	Adjoint switch (> 0 for adjoint problem).
NGPQT1 NGPQT NGPQT3	Problem-dependent energy group limits (see Appendix A).
NGPQTG	Group number of lowest energy gamma-ray group to be treated.
NGPQTN	Group number of lowest energy neutron group to be treated.
NITS	Number of batches to be run.
NLAST	Last cell in blank common used by random walk package.
NLEFT	Number of cells in blank common available to user.
NMGP	Number of primary particle energy groups.
NMTG	Total number of energy groups.
NSTRT	Number of source particles for each batch.

TABLE VIII.

Definitions of Variables in Common GØMLØC

Variable	Definition
KMA	Starting location for the array MA containing integer data for each code zone.
KFPD	Starting location for the array FPD containing real data for each code zone.
KLCR	Starting location for the array LØCREG(I) which contains the starting location in the MA array for the Ith code zone data.
KNBD	Starting location for the array NUMBØD(I) which contains the number of bodies for the Ith code zone.
KIØR	Starting location for the array IRØR(I) which contains the index of the corresponding input zone for the Ith code zone.
KRIZ	Starting location for the array MIRZ(I) which contains the index of the MØRSE region corresponding to the Ith geometry input zone.
KRCZ	Starting location for the array MRCZ(I) which contains the index of the MØRSE region corresponding to the Ith geometry code zone.
KMIZ	Starting location for the array MMIZ(I) which contains the index of the MØRSE media corresponding to the Ith geometry input zone.
KMCZ	Starting location for the array MMCZ(I) which contains the index of the MØRSE media corresponding to the Ith code zone.
KKR1	Starting location for the array KR1(L) contains the first code zone which was made from the Lth input zone.
KKR2	Starting location for the array KR2(L) contains the last code zone which was made from the Lth input zone.
KNSR	Starting location for array NSØR which contains the code zones in which source particles have been found.

TABLE VIII. (Cont'd.)

Variable	Definition
KVØL	Starting location for the array VNØR(I) which contains the volume for MØRSE region (I).
NADD	Starting location for the geometry data length and changed in JØMIN to the total number of words required for geometry data.
LDATA	Length of the integer data in the MA array excluding the words set aside for zone search information.
LTMA	Total length of the MA array.
LFPD	Length of the FPD array
NUMR	Number of code produced zones.
IRTRU	Number of input zones.
NUMB	Number of bodies.
NIR	Number of MØRSE geometry regions.

TABLE IX.

Definitions of Variables in Common LØCSIG

Variable	Definition
ISTART	Starting location for the total cross-section vector for the first medium.
ISCCØG	Starting location for the scattering cross-section vector for the first medium.
INABØG	Starting location for the non-absorption vector for the first medium.
IGABØG	Starting location for the gamma-ray production vector for the first medium.
IFPØRG	Starting location for $\nu\Sigma_f$, the fission neutron production vector for the first medium.
IFNGP	Starting location for the primary-secondary transfer probability matrix.
IFSPØG	Starting location of the primary downscatter probability matrix.
IDSGØG	Starting location of the secondary downscatter probability matrix.
IPRBNG	Starting location of the primary scattering angle probability matrix.
IPRBGG	Starting location of the secondary scattering angle probability matrix.
ISCANG	Starting location of the primary scattering angle matrix.
ISCAGG	Starting location of the secondary scattering angle matrix.
ISPØRG	Size of storage needed for each medium, not including Legendre coefficients.

TABLE IX. (Cont'd.)

Variable	Definition
ISPØRT*	Starting location for temporary storage of downscatter matrix.
INPBUF	Starting location for temporary storage of the P_0 table.
ISIGØG	Starting location for temporary storage of total cross section for element 1.
INFPØG*	Starting location for temporary storage of $\nu\Sigma_f$ for element 1.
IABSØG	Starting location for temporary storage of downscatter matrix for P_L coefficients (primary groups, element 1).
ITØTSG*	Total storage required for temporary storage.
NGP	The number of primary groups to be treated.
NDS	Number of downscatters for NGP (usually equal to NGP).
NGG	Number of secondary groups to be treated.
NDSG	Number of downscatters for NGG (usually equal to NGG).
INGP	Number of groups for which cross sections are to be input.
INDS	Number of downscatters for the INGP groups.
NMED	Number of media for which cross sections are to be stored--should be same as MEDIA as read on Card B of MØRSE input.
NELEM	Number of elements for which cross sections are to be read.

* If Legendre coefficients are to be restored, then:

INFPØG - Redefined by JNPUT as number of locations required for each coefficient (both primary and secondary).

ITØTSG - Redefined by JNPUT as total storage required for all coefficients for each medium.

ISPØRT - Redefined by JNPUT as starting location of P_1 coefficient for primary groups for medium 1.

TABLE IX. (Cont'd.)

Variable	Definition
NMIX	Number of elements times density operations to be performed.
NCØEF	Number of coefficients, including P_0 .
NSCT	Number of discrete angles (usually NCØEF/$2_{Integral}$).
NTS	Number of downscatters for combined primary and secondary groups (usually equal to NTG).
NTG	Total number of groups (primary + secondary) = NGP + NGG.
NDSNGP	The number of locations needed for the downscatter matrix for the primary particles.
NDSNGG	The number of locations needed for the downscatter matrix for the secondary particles.
IADJ	Same as IADJM.
NME	Indicator for stripping gamma rays from a coupled neutron gamma-ray cross-section set--set equal to number of neutrons groups + 1.
LØC	Same as LØCEPR.
INGS	Starting location of the indices for starting location of the downscatter vector for each group for primary particles.
INSG	Starting location of the indices for starting location of the downscatter vector for each group for secondary particles.
I1, I0	Input and output logical unit numbers.
KKK	A running index of the number of cross sections that have already been read in (used in checking the element numbers obtained from tape).
IXTAPE	Logical tape number of the multigroup cross-section tape if > 0, or logical tape number of the processed cross-section tape if < 0.
IDEL	Starting location for element identifiers which determine the element cross sections to be read from tape.

TABLE IX. (Cont'd.)

Variable	Definition
ITEML	Amount of storage for primary and secondary group downscatters per element.
ITEMG	Starting location for temporary storage of downscatter matrix for P_L coefficients (secondary groups) for element 1.
IRSG	Starting location of the mixing parameters.
IRDSG	Switch to print the cross sections and to test the card sequence as they are read if > 0 (test card sequence only if = 0, and does neither if < 0).
ISTR	Switch to print cross sections as they are stored if > 0.
IPRIN	Switch to print angles and probabilities if > 0.
IFMU	Switch to print intermediate results of μ's calculation if > 0.
IMØM	Switch to print moments of angular distribution if > 0.
IDTF	Switch to signal that input format is DTF-IV format if > 0; otherwise, ANISN format is assumed.
ISTAT	Flag to restore Legendre coefficients for next-flight estimates if > 0.
IPUN	Switch to print results of bad Legendre coefficients if > 0.
NUS	Number of groups of upscatter.
NGN	Not used presently.
IHT	Location of Σ_T in the cross-section table, currently set to 3 in XSEC.
INUS	Starting location for the upscattering vector for the first medium.
INUSN	Number of impossible first moments found in the cross sections.
INGN	Starting location for the photoneutron production vector for the first medium (not used at present--same as INUS).

TABLE IX. (Cont'd.)

Variable	Definition
INGNP	Starting location for the secondary-primary transfer probability matrix (not used at present--same as INUS).
INNN	Starting location for the array of the number of down-scatter groups for each primary group.
IGGG	Starting location for the array of the number of down-scatter groups for each secondary group.

TABLE X.

Definitions of Variables in Common GTSC1

Variable	Definition
NPT(16)	Number of points per supergroup for each medium. (Only 16 media are allowed.) Redefined at end of GTSCT as the total storage required for each cross section for each medium.
NXSECT(17)	Starting location of point cross sections for each medium.
NDSGP	Total storage required per supergroup for all media.
I1, I0	Input and output logical unit numbers.
IØ6RT	Logical tape number of Ø6R point cross-section tape.
IGQPT	Last MØRSE multigroup for which point cross sections will be used (≤ NMGP).
NEGPS	Number of supergroup boundaries.
ESPD(100)	Energy boundaries (in eV) of the supergroups (only 100 supergroups are allowed).
NNIC	Starting location of the point cross sections.
NXPM	Number of cross sections per medium = 1 if total only = 2 if total + scattering = 3 if total, scattering, and ν*fission.
NINC	Index for locating point cross sections. Used as flag in FISGEN to determine if transport process has begun.

TABLE XI.

Definitions of Variables in Common PDET

Variable	Definition
ND	Number of detectors (≥ 1).
NNE	Number of neutron energy bins.
NE	Total number of energy bins.
NT	Number of time bins.
NA	Number of angle bins.
NRESP	Number of energy response functions (≥ 1).
NEX	Number of extra arrays of length NMTG to be set aside.
NEXND	Number of extra arrays of length ND to be set aside.
NEND	NE*ND.
NDNR	ND*NRESP.
NTNR	NT*NRESP.
NTNE	NT*NE.
NANE	NA*NE.
NTNDNR	NT*ND*NRESP.
NTNEND	NT*NE*ND.
NANEND	NA*NE*ND.
LØCRSP	Location of cell zero of response functions.
LØCXD	Location of cell zero of detector positions.
LØCIB	Location of cell zero of energy bins.
LØCCØ	Location of cell zero of angle (cosine) bins.
LØCT	Location of cell zero of time bins.
LØCUD	Location of cell zero of array UD.
LØCSD	Location of cell zero of array SD.
LØCQE	Location of cell zero of array QE.
LØCQT	Location of cell zero of array QT.

TABLE XI. (Cont'd.)

Variable	Definition
LØCQTE	Location of cell zero of array QTE.
LØCQAE	Location of cell zero of array QAE.
LMAX	Last cell used in blank common.
EFIRST	Upper energy limit of first energy bin.
EGTØP	Upper energy limit of first gamma-ray bin.

TABLE XII.

Definitions of Variables in Common FISBNK

Variable	Definition
MFISTP	Index to indicate that fissions are to be treated if > 0.
NFISBM	Location in blank common of cell zero of the fission bank.
NFISH	Number of fissions produced in the previous batch (generation).
FTØTL	Total fission weight from all collisions in this batch (generation).
FWATE	Total weight of fission neutrons stored in bank in this batch (generation).
WATEF	Weight of fission neutron stored in bank.

TABLE XIII.

Definitions of Variables in Common GEØMC
as found in Ø5R Geometry

Variable	Definition
X2, Y2, Z2	Coordinates at tentative end-of-flight or if the trajectory is in an internal void; X2, Y2, Z2 are the direction cosines of the trajectory.
X1, Y1, Z1	Starting coordinates for the particle.
ETA	Number of mean free paths to be traversed if flight goes to X2, Y2, Z2.
ETAUSD	Number of mean free paths actually traversed after the call to GEØM.
IBLZ	An index to the medium number for the special geometry packages. For GENERAL GEØM, IBLZ is a packed work giving the block and zone of the end of flight.
IBZN	A dummy variable.
MARK	A flag set by GEØM indicating the results of the trajectory calculation = 1 for completed flight = 0 for boundary crossing = -1 for escape = -2 for entering an internal void.
NMED	Medium number at end of flight or of medium about to be entered at a boundary crossing.
NREG	Region number at end of flight; not set at boundary crossings.

TABLE XIV.

Definitions of Variables in Common PAREM
as found in Combinatorial Geometry

Variable	Definition
XB(3)	Coordinates of the starting point of the present path.
WB(3)	Direction cosines of particle trajectory. Equal to U, V, and W.
IR	Combinatorial zone of present particle position.
WP(3)	Temporary storage of WB(3).
XP(3)	Temporary storage of XB(3).
IDBG	Set non-zero to initialize a debug printout.
IRPRIM	Next region to be entered after a call of G1.
NASC	Body number of last calculated intersection. Set negative to indicate source or collision point not on a body surface.
LSURF	Surface of body NASC where intersection occurred. Positive if particle is entering the body and negative when exiting.
NBØ	Body number and a sign used to define zones. Input in zone description as positive when zone is contained in body and as negative if zone is outside body.
LRI	Entry surface calculated in GG.
LRØ	Exit surface calculated in GG.
RIN	Distance to entry calculated in GG.
RØUT	Distance to exit calculated in GG.
KLØØP	Trajectory index of present path incremented in G1.
LØØP	Index of last trajectory calculated for body NBØ. If LØØP is equal to KLØØP, GG returns immediately with old values of RIN, RØUT, LRI, and LRØ.
ITYPE	Body type of body NBØ (indicates BØX, SPH, etc.).

TABLE XIV. (Cont'd.)

Variable	Definition
PINF	Machine infinity.
NØA	Not used.
DIST	Distance from XB(3) to present point.

TABLE XV.

Definitions of Variables in Common ØRGI*

Variable	Definition
MARK	Set 1 in G1 if trajectory end point is reached before next intersection. Otherwise set to 0.
DIST0	Distance from point XB(3) to next scattering point. Used in G1 to avoid calculating the next zone if a scattering event occurs before the intersection.
NMEDG	Zone number IR from a LØØKZ call. Stored in BLZNT by MSØUR.

* Note: Variabl names are not the same in all routines.

TABLE XVI.

Definitions of Random Walk Variables in Blank Common

Mnemonic Variable Name	Definition
ENER(IG)	Upper energy boundary of group IG (in eV).
VEL(IG)	Velocity corresponding to the mean energy for neutron groups and the speed of light for gamma-ray groups (in cm/sec).
FS(IG)	Unbiased source spectrum--unnormalized fraction of source particles in each energy group--transformed to c.d.f. by SØRIN.
BFS(IG)	Biased source spectrum--relative importance of each energy group--transformed to biased c.d.f. by SØRIN.
WTHI(IG, NREG)	Weight above which splitting is performed (vs. group and region).
WTLØ(IG, NREG)	Weight below which Russian roulette is performed (vs. group and region).
WTAV(IG, NREG)	Weight to be assigned Russian roulette survivors (vs. group and region).
PATH(IG, NREG)	Exponential transform parameters (vs. group and region).
NSPL(IG, NREG)	Splitting counter (vs. group and region).
WSPL(IG, NREG)	Weight equivalent to NSPL.
NØSP(IG, NREG)	Counter for full bank when splitting was requested (vs. group and region).
WNØS(IG, NREG)	Weight equivalent to NØSP.
RRKL(IG, NREG)	Russian roulette death counter (vs. group and region).

TABLE XVI. (Cont'd.)

Mnemonic Variable Name	Definition
WRKL(IG, NREG)	Weight equivalent to RRKL.
RRSU(IG, NREG)	Russian roulette survival counter (vs. group and region).
WRSU(IG, NREG)	Weight equivalent to RRSU.
INIWHI (IG, NREG)	Initial values of WTHI array.
INIWLØ(IG, NREG)	Initial values of WTLØ array.
INIWAV(IG, NREG)	Initial values of WTAV array.
FWLØ(NREG)	Weights to be assigned to fission daughters (vs. region).
INIFLØ(NREG)	Initial values of FWLØ.
GWLØ(IG, NREG)	Weights to be assigned to secondary particles (vs. group and region).
EPRB(IG, NREG)	Relative importance of energy groups after scattering (vs. group and region).
NSCT(IG, NREG)	Number of real scatterings (vs. group and region).
WSCT(IG, NREG)	Weight equivalent to NSCT.
NALB(IG, NREG)	Number of albedo scatterings (vs. group and region).
WALB(IG, NREG)	Weight equivalent to NALB.
NFIZ(IG, NREG)	Number of fissions (vs. group and region).

TABLE XVI. (Cont'd.)

Mnemonic Variable Name	Definition
WFIZ(IG, NREG)	Weight equivalent to NFIZ.
NGAM(IG, NREG)	Number of secondary productions (vs. group and region).
WGAM(IG, NREG)	Weight equivalent to NGAM.
NSCA(IMED)	Scattering counter (vs. cross-section medium).
FISH(IG, IMED)	Probability of generating fission neutron (vs. group and medium).
FSE(IG, IMED)	Source spectrum for fission-induced neutrons for each group--input as frequency of group IG.
GMGN(IG, IMED)	Probability of generating secondary particle (vs. group and medium).

IV. THE PICTURE PROGRAM

This program is used to obtain printer plots of two-dimensional slices through the three-dimensional geometry described by the combinatorial geometry (CG) package. The Ø5R geometry version[10] has been modified for use with CG and some minor changes to the program have been made. It is now possible to obtain a picture the full width of a computer page and the logic for determining the medium and/or region number has been changed so that a test of the geometry ray tracing capability is also made. That is, instead of just testing the routine LØØKZ, this modification forces the geometry package to execute the routines and logic used during the random walk process. This program has been very useful in debugging geometry input. It is suggested that any complex geometry be tried in this program before using it in the MØRSE code. A sample problem is provided. Besides the geometry package, a main program, PICTURE, and two routines, MESH and PRINT, are required. A discussion of these routines is included and a definition of the variables in common, PICT, is given in Table XVII.

Preceding page blank

4.1 Program PICTURE

PICTURE is the executive routine for the PICTURE program and reads in the input, calculates the coordinates of the picture to be plotted and controls the calls to other routines. There are several different ways in which a two-dimensional slice through the geometry may be obtained. These different options are discussed in Appendix B with discussions of the input. The characters to be printed for corresponding media or regions may be changed by altering the values in ATABLE as given in the data statement.

Subroutines called: JØMIN, PRINT

Commons required: PICT

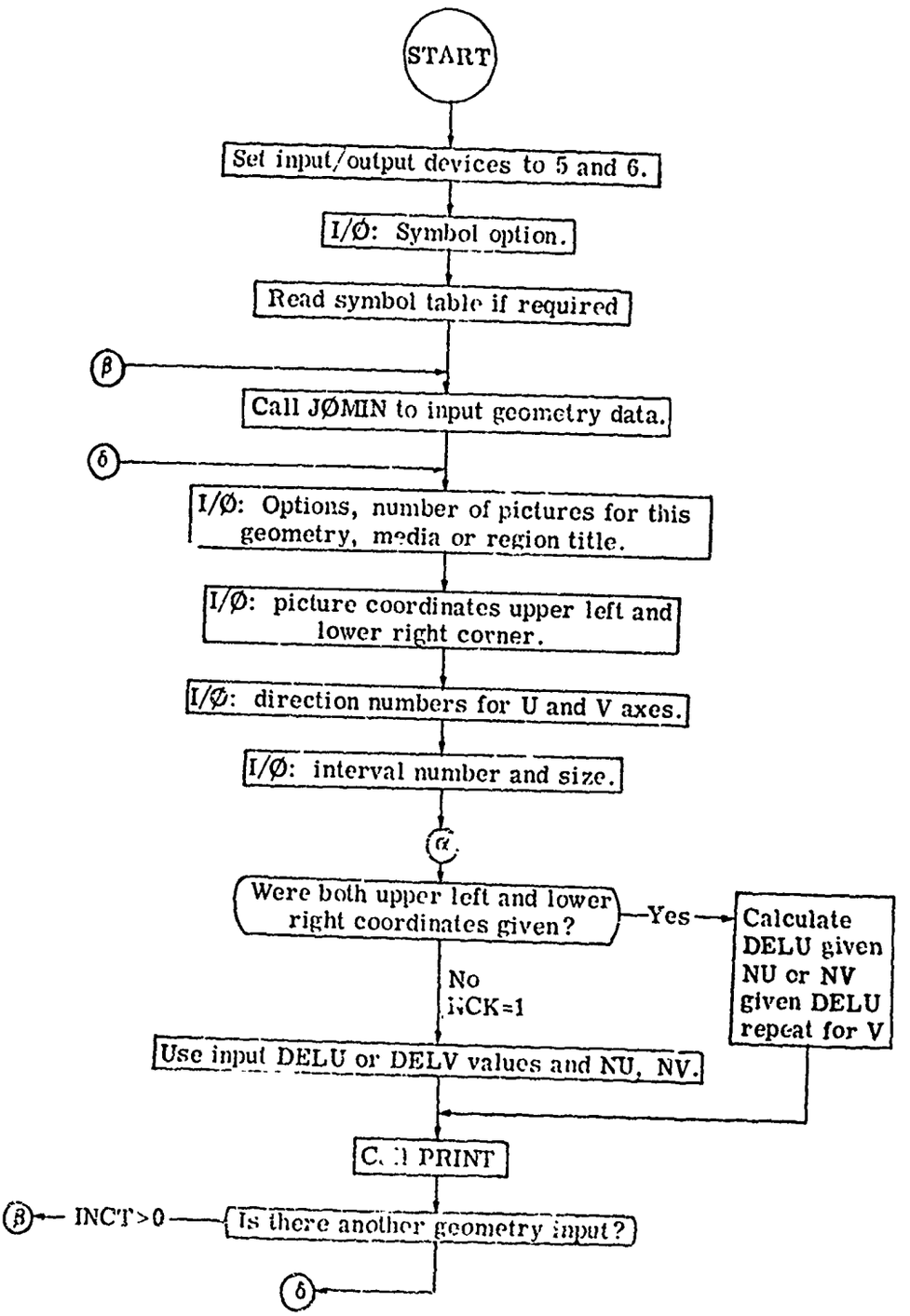
Variables required: Several input cards are read.

Variables changed: All variables in common PICT.

Significant internal variables:

INT	-	input logical unit,
IØT	-	output logical unit,
NADD(1)	-	first location in blank common for storage of geometry data,
NCK	-	flag to indicate which set of input options was used to define the two-dimensional slice.

Program PICTURE



4.2 Subroutine PRINT (KXX, KYY, ATABLE)

This routine controls the printing of the picture. First it is determined if the picture is to be more than one "page" (130 columns) wide and then for each line of the picture on the first page the information to be printed is determined by calling MESH and printed with a 130A1 format. The next lines are then calculated and printed until that "page" is finished. Note that a page refers to width not length. Thus, as much detail may be obtained as necessary in both directions by piecing together the output.

Called from: PICTURE

Subroutine called: MESH

Commons required: PICT

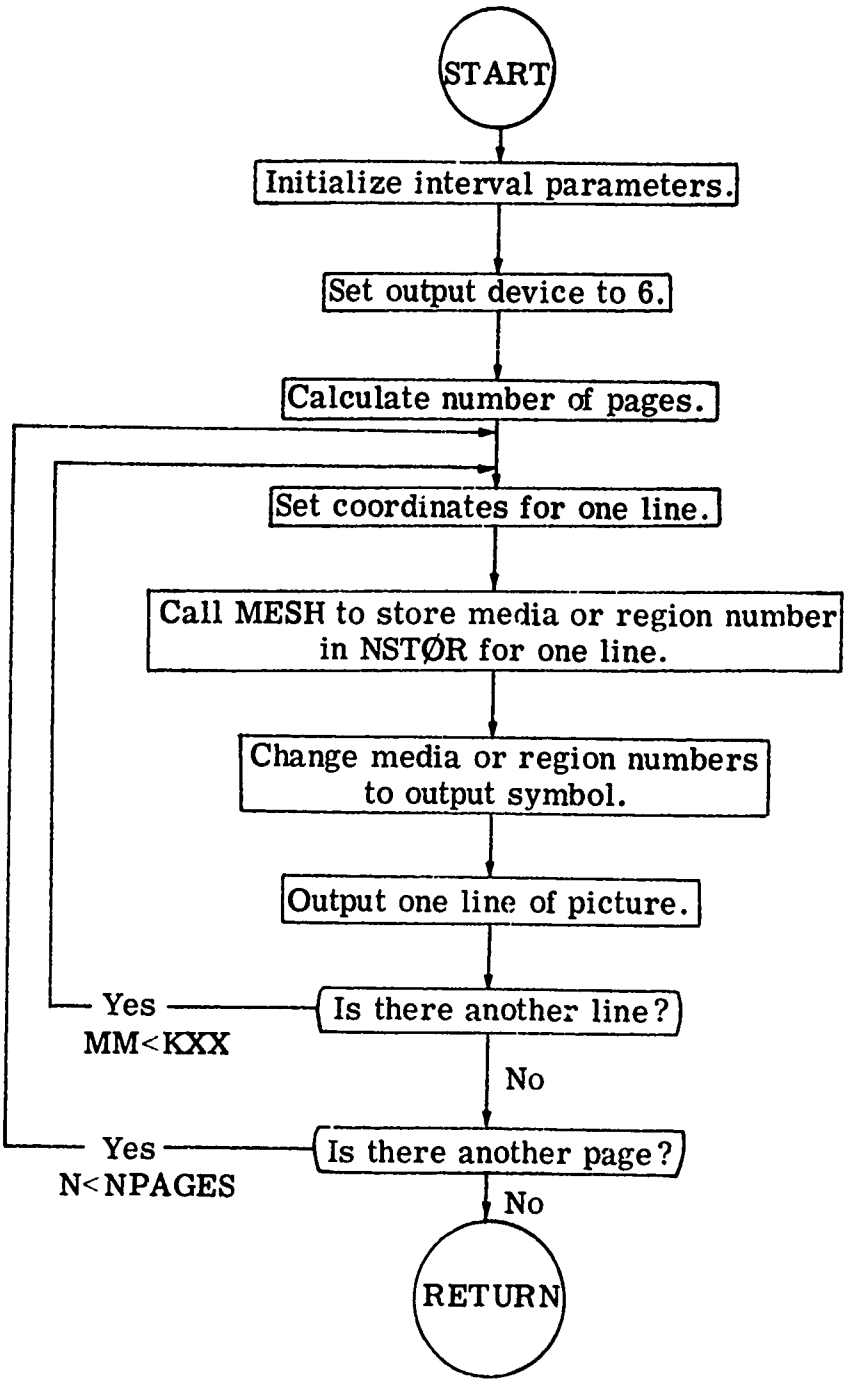
Variables required:

KXX	-	number of intervals in the U direction (direction of paper movement through the printer),
KYY	-	number of intervals in the V direction (line),
ATABLE	-	table of characters to be printed.

Significant interval variable:

NPAGES	-	number of subpictures required to cover the width of the total picture,
IØT	-	output logical unit,
NV	-	number of characters per line (characters/page width).

Subroutine PRINT



4.3 Subroutine MESH (XS, YS, ZS, NV)

Subroutine MESH is used by the PICTURE package to set up one line of print in the array NSTØR. Both for efficiency and to debug the combinatorial geometry package, this version has been modified to work exactly like particle tracking. LØØKZ is first called to determine the zone of the first grid point. A trajectory to the last grid point is then initialized, and successive calls to G1 "track" a particle to the last point, setting the region of each grid point in NSTØR. By setting IRG negative, zero, or positive, either NREG, IR, or NMED will be stored in the print array NSTØR.

Called from: PRINT

Subroutines called: LØØKZ, G1

Commons required: PICT, GØMLØC, PAREM, ØRGI, blank.

Variables required:

XS, YS, ZS	-	coordinates of first grid point,
NV	-	number of grid points,
DELV	-	distance between grid points,
IRG	-	flag to print NREG, IR, or NMED if IRG is negative, zero, or positive.

Variables changed: NSTØR - print array.

Significant internal variables:

J	-	grid point index,
ISTØR	-	value to be stored in NSTØR between successive boundary crossings.

Subroutine MESH

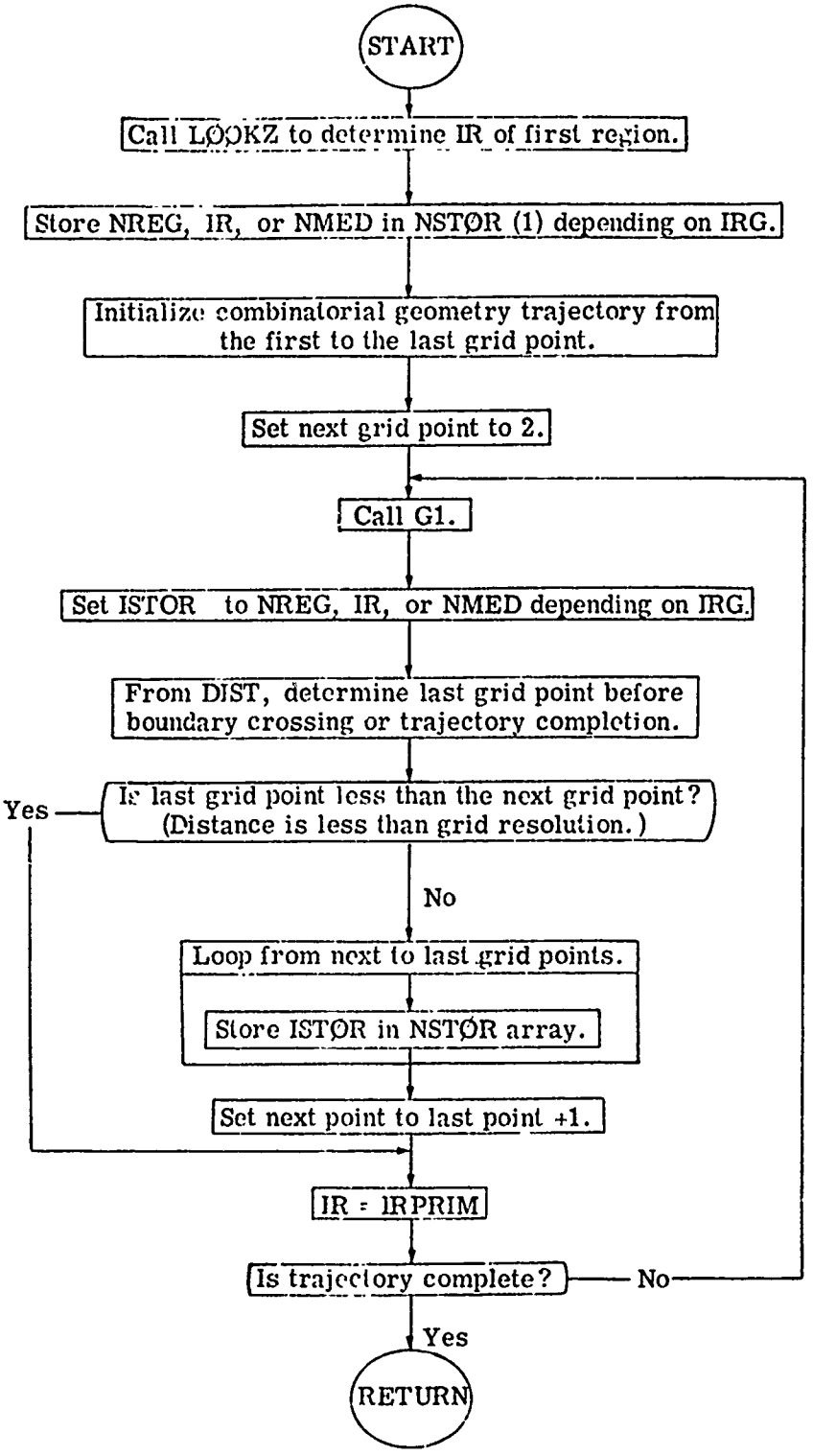


TABLE XVII.

Definition of Variables in Common PICT*

Variable	Definition
DELU	The increment in geometry units between lines in the picture in the U direction.
DELV	The increment in geometry units between lines in the picture in the V direction.
X0, Y0, Z0	The coordinates in geometry units defining the first point (upper left hand corner) of the picture.
XU, YU, ZU	The length (in direction U) of the picture in geometry units.
XV, YV, ZV	The width (in direction V) of the picture in geometry units.
NSTØR(130)	An array used to store the medium, region, or zone number for one line of the picture.
IRG	A flag indicating that region, zone, or medium geometry parameter should be printed if IRG is negative, zero, or positive, respectively.

* Used in the PICTURE program.

V. MØRSE SAMPLE PROBLEMS

5.1 Sample Problem Number 1

This sample problem is similar to that used previously[7] which calculates neutrons and secondary gamma rays in infinite homogeneous air. The source is a 12.2- to 15-MeV neutron source. Two types of estimators are used in the analysis of the particle histories. First, a boundary-crossing estimator is used for five spherical shells. (This part of the problem is similar to the MØRSE sample problem.) Second, a collision density estimator is used in RELCØL for nine detector regions (detectors 6-14). By choosing the spherical shells as the midpoints of the regions, an internal check is provided. Also, by choosing the geometry regions to correspond to the collision density detectors, the results from SAMBØ can be compared with results from ENDRUN. With the inclusion of the combinatorial geometry package, an additional estimator based on track length/unit volume has been added with results also output in ENDRUN. Both the collision density and track length per unit volume estimators are obtained for the geometry regions without specific coding.

Since the collision density estimator in RELCØL scores $4\pi R_i^2 \varphi$ there will be some differences between these estimates and those from the collision density estimator output in ENDRUN which scores $4\pi \langle R \rangle^2 \varphi$. (The symbols $\langle\ \rangle$ denote an average over a spatial region.)

Table XVIII illustrates the comparison for one of several sets of answers as calculated by the various estimators. Descriptions of the user routines INSCØR, RELCØL, and BDRYX required for the sample problem are given.

TABLE XVIII.

Comparison of Results of Sample Problem 1

Detector No.	Type and Location of Detector	Result: Response #1 $4\pi R^2$ fluence top group	Result: Group 1 fluence/eV*
3	Boundary crossing at 450 meters.	8.19(-2)	2.92(-8)*A = 8.19(-2)
9	Collision density for volume between 400-500 meters.	7.59(-2)	2.71(-8)*A = 7.59(-2)
Region 4	Volume between 400-500 meters (ENDRUN results).	Real Collision Region 4 Top Group	1.11(-18)*A*B = 7.76(-2)
	Volume between 400-500 meters (ENDRUN results).	Track Length Region 4 Top Group	1.18(-18)*A*B = 8.36(-2)

* A = width of the top group 2.8(+6) eV.
B = $4\pi \langle R \rangle^2$ for 4th region = 2.5(+10).

5.1.1 Subroutine INSCØR - Sample Problem Number 1

This user routine is used to calculate the volume of each of the detector regions (detectors 6-14) in the sample problem. The reciprocal of the volume is stored in the array in blank common set aside for FACT, a detector-dependent normalization. Because two types of estimators are used in the problem, normalization values of 1 are stored for the boundary detectors. The detector number NDC for the last boundary crossing detector (in this case, 5) must be input.

Called from: SCØRIN

Subroutines called: None

Commons required: PDET, USER, DETCUT, blank.

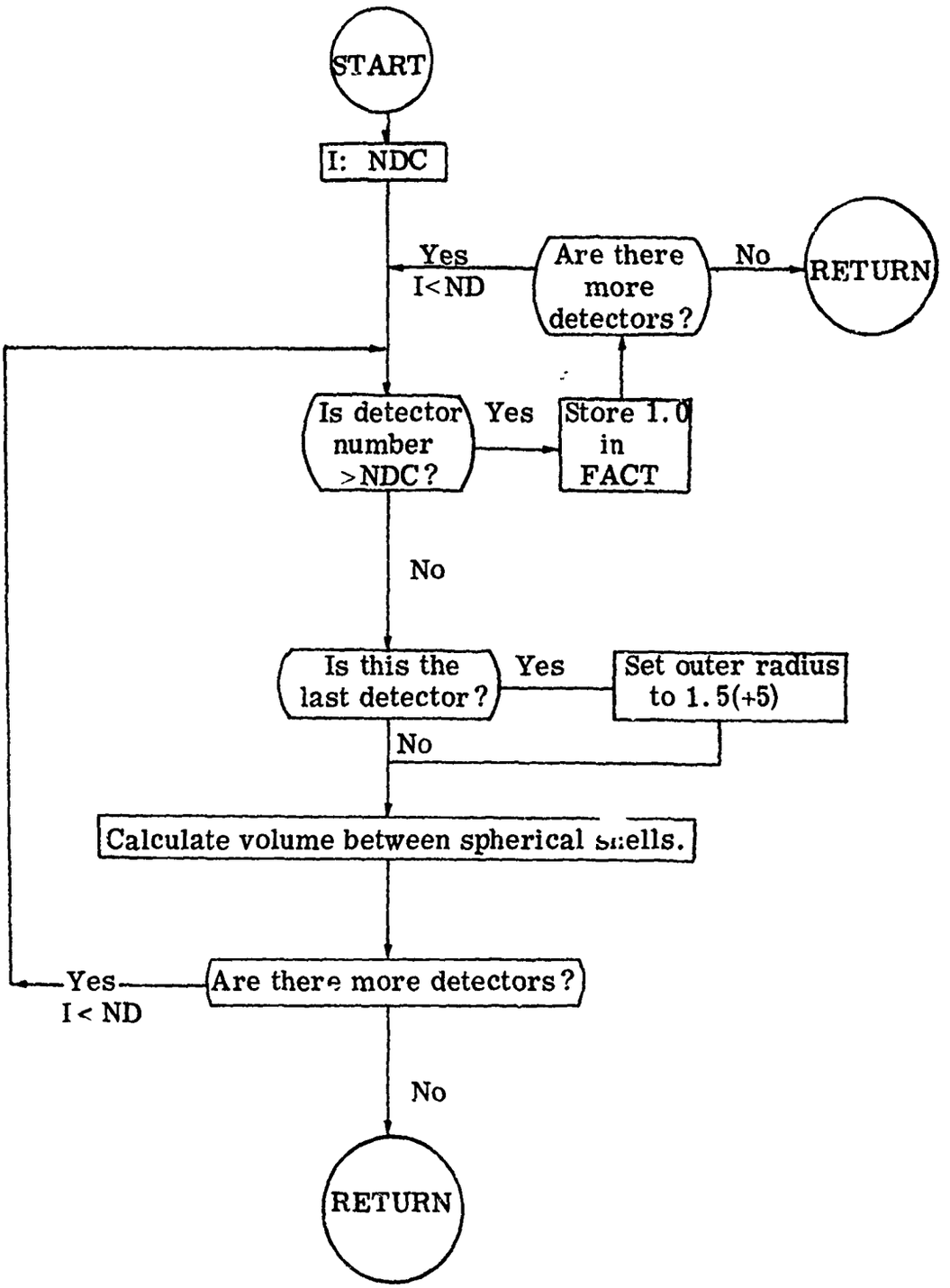
Variables required:

NDC - the number of the last boundary crossing detector. Detector radii from blank common.

Variables changed:

Normalization array in blank common.

Subroutine INSCØR



5.1.2 Subroutine RELCØL - Sample Problem Number 1

This subroutine estimates the fluence from the density of collisions. For comparison purposes, $4\pi R_i^2$ times the fluence is stored where R_i is the radius to each collision site. Thus, the contribution is

$$4\pi P_i^2 \frac{WTBC}{\Sigma_T} ,$$

where

$R_i = X_i^2 + Y_i^2 + Z_i^2$,

WTBC = particle weight before collision,

Σ_T = cross section for the group (IGØ) of the incoming particle.

A detector number from NDC + 1 to ND is chosen depending on the value of R_i.

Called from: BANKR(5)

Subroutines called: FLUXST, NSIGTA

Commons required: PDET, NUTRØN, USER, DETCUT

Variables required:

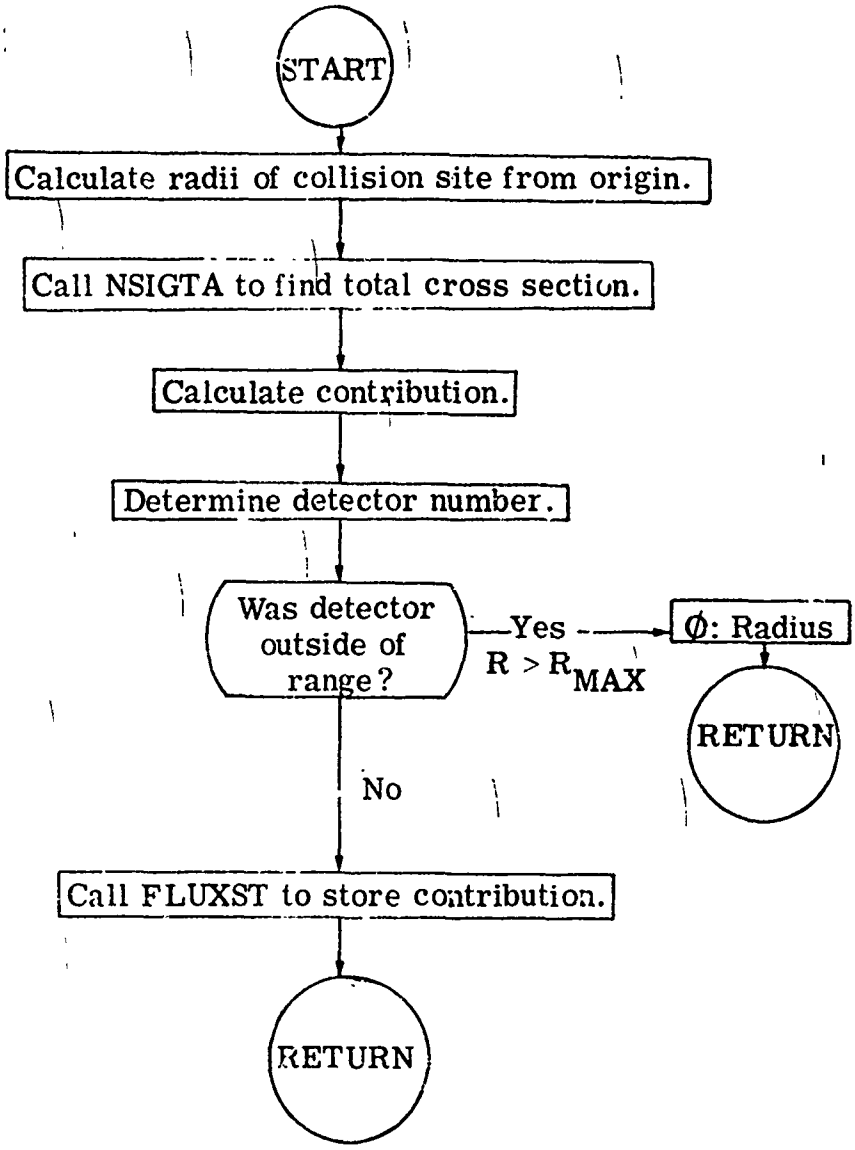
X, Y, Z, IGØ, NMED, AGE, WTBC from NUTRØN common TSIG and detector radii.

Significant internal variables:

R - radial distance to collision site,

CØN - fluence estimate.

Subroutine RELCØL



5.1.3 Subroutine BDRYX - Sample Problem Number 1

This subroutine is called whenever the particle in the random walk encounters a change in geometry media. If the source-to-collision distance corresponds to a detector position, the reciprocal of the cosine of the angle from the radius vector is used as a fluence estimate.

Called from: BANKR(7)

Subroutines called: ERRØR

Commons required: USER, NUTRØN, DET

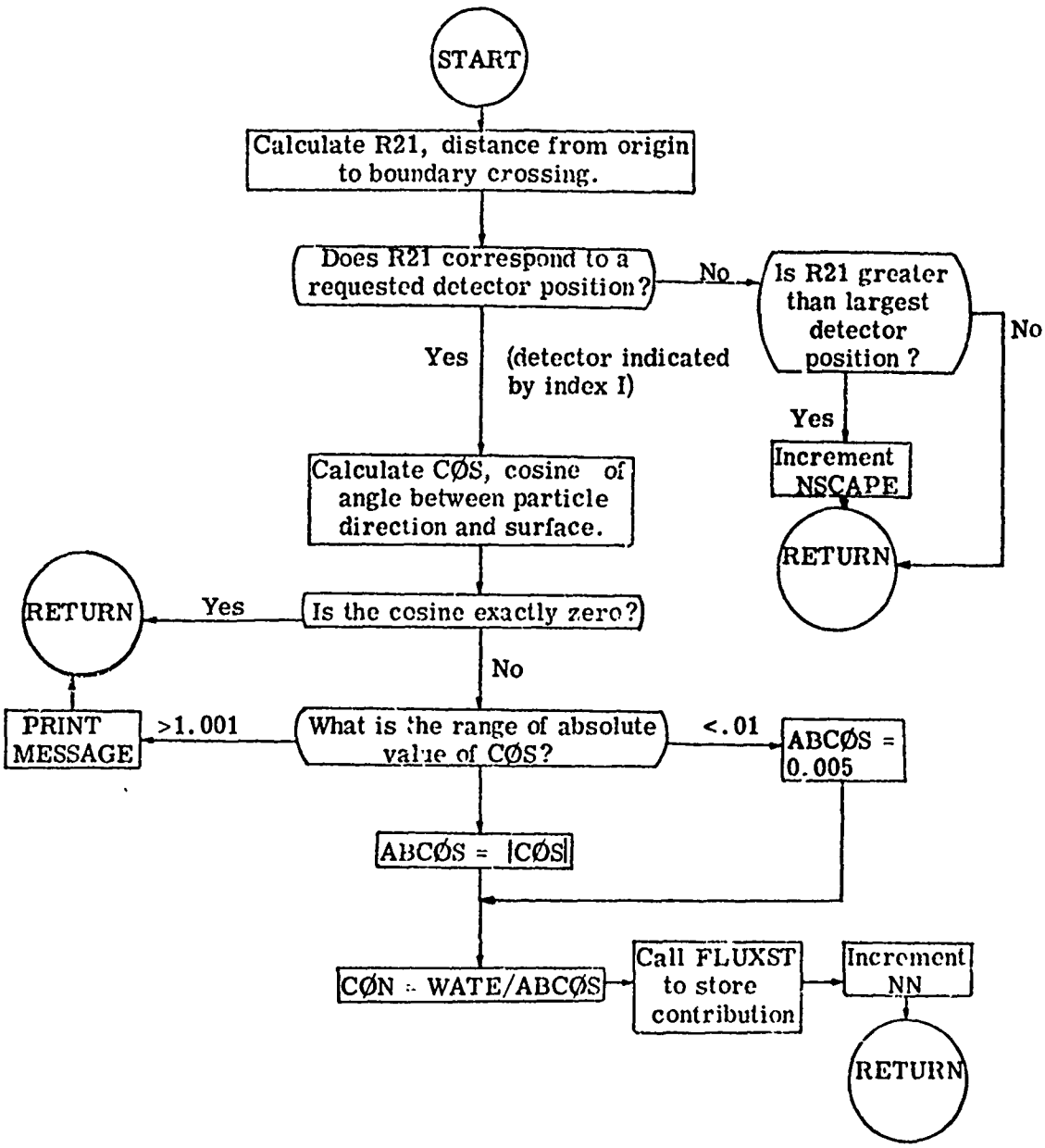
Variables required:

X, Y, Z, U, V, W, WATE (from common NUTRØN).

Significant internal variables:

R21	-	radial distance to boundary crossing,
R2	-	99% of R21,
R22	-	101% of R21,
CØS	-	cosine of angle between particle direction and radius vector,
ABCØS	-	absolute value of CØS,
CØN	-	fluence estimate.

Subroutine BDRYX



5.2 Sample Problem Number 2

This sample problem[11] is used to illustrate the next event estimator. The neutron and secondary gamma ray spectra at several point detectors located 10^4 cm from a polyethylene slab is calculated. The source is a beam of neutrons with a fission spectrum energy distribution. The use of the point detector routines to estimate the energy dependent fluence is illustrated and in particular, for secondary gamma rays the contribution of the uncollided or first flight gamma rays are scored in subroutine SGAM. The estimates from real collisions are made in subroutine RELCØL. Results are recorded in Table XIX in units of particles cm^2 source neutron.

TABLE XIX.

Results of Last Flight Estimator
Sample Problem

Detector	Neutrons/cm^2 Per Source Neutron	Gammas/cm^2 Per Source Neutron
1	6.24(-11)	2.36(-11)
2	6.44(-10)	4.82(-11)
3	1.32(-9)	4.93(-11)
4	1.56(-9)	4.94(-11)
5	6.92(-1)	2.15(-3)

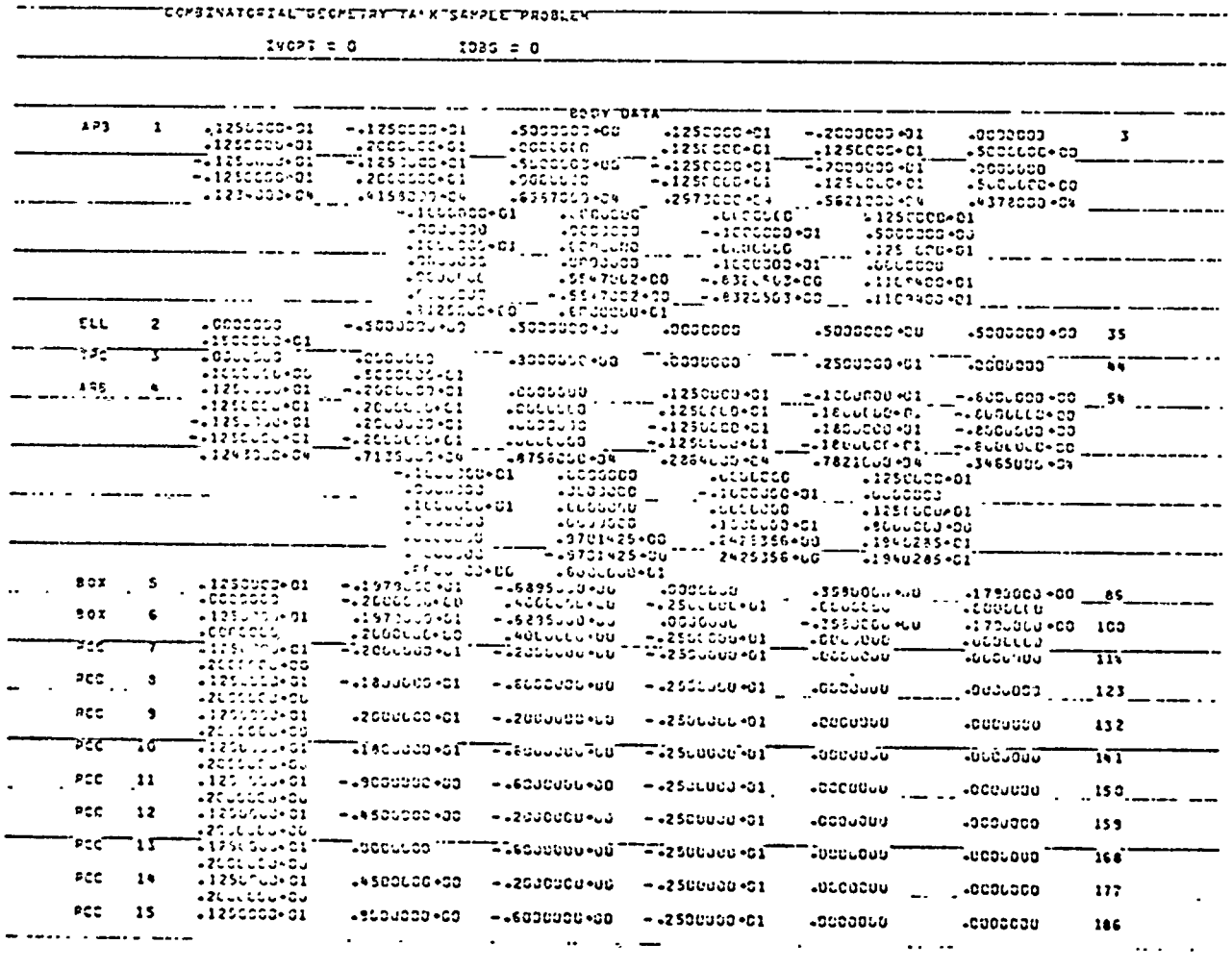
5.3 Sample Problem Number 3

This sample problem serves two purposes by (1) demonstrating the use of the PICTURE code, and (2) verifying the tracking capability of the combinatorial geometry package in a moderately complex three-dimensional system. In the PICTURE program, subroutine G1 is called from MESH in the exact manner that it is called from GØMST during a MØRSE calculation. As such, PICTURE works by tracking rays and thus can be used to debug a combinatorial geometry input.

The sample problem itself is a picture of a tank. The tank model itself was constructed purely as an illustration of the combinatorial geometry and is in no way accurate or detailed. Notice that the use of both the ØR operator and the ARB body is demonstrated.

TABLE XX.

Output from PICTURE Sample Problem



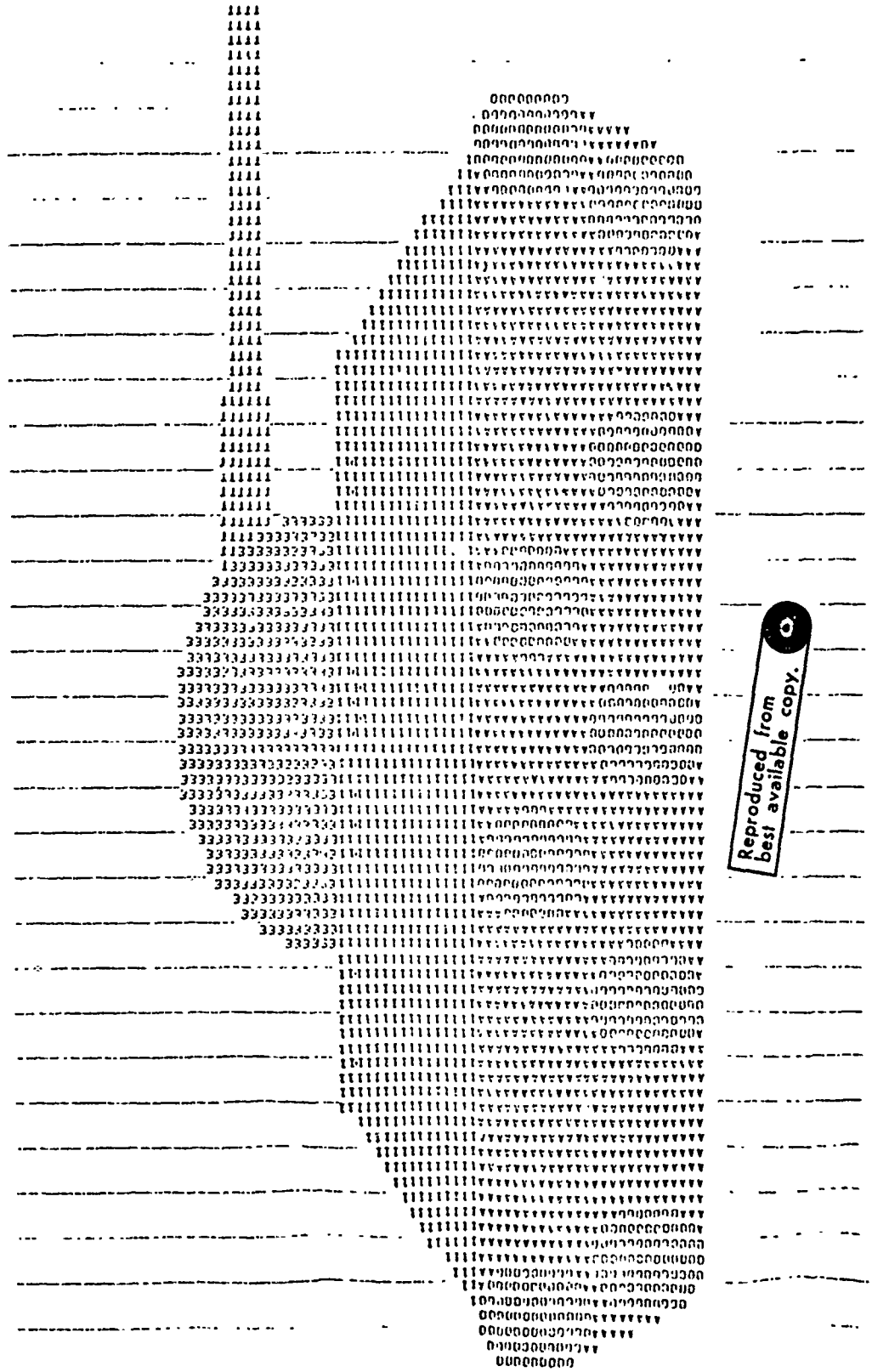
Reproduced from best available copy.

TABLE XX. (Cont'd.)

```
                .2000000+00
 PPP    16   -.1000000+02    .1000000+02   -.1000000+02    .1000000+02   -.1000000+02    .1000000+02   395
 END    17    .0000000       .0000000       .0000000       .0000000       .0000000      -.0000000      203
NUMBER OF BODIES        16
LENGTH OF FPD-ARRAY   208

                                                  INPUT ZONE DATA
   1    0      1      0      0      0      0      0      0      0      0  Z  1
   2    0      2     -1      0      0      0      0      0      0      0  Z  2
   3    0      3     -2      0      0      0      0      0      0      0  Z  3
   4    0OR    4     -7     -8     -9    -10    -11    -12    -13    -14  Z  4
        0    -15OR    5     -7    -8OR    6     -9    -10      0      0  Z  5
   5    0OR    7OR    8OR    9OR   10OR   11OR   12OR   13OR   14OR   15  Z  7
   6    0     16     -1     -2     -3     -4     -5     -6     -7     -8  Z 16
        0     -9    -10      0      0      0      0      0      0      0  Z 17
 END    0      0      0      0      0      0      0      0      0      0  Z 17
NUMBER OF INPUT ZONES      6
NUMBER OF  CODE ZONES     16
LENGTH OF INTEGER ARRAY  353

CODE ZONE    INPUT ZONE    ZONE DATA LOC.    NO. OF BODIES    REGION NO.    MEDIA NO.
     1            1            113                1              1            1
     2            2            118                2              1            2
     3            3            127                2              1            3
     4            4            135               10              1            4
     5            4            177                3              1            4
     6            4            190                3              1            4
     7            5            203                1              1            5
     8            5            208                1              1            5
     9            5            213                1              1            5
    10            5            218                1              1            5
    11            5            223                1              1            5
    12            5            228                1              1            5
    13            5            233                1              1            5
    14            5            238                1              1            5
    15            5            243                1              1            5
    16            6            248               11              1         1000

   I    KR1(I)    KR2(I)
   1      1         1
   2      2         2
   3      3         3
   4      4         6
   5      7        15
   6     16        16

        MORSE REGION IN INPUT ZONE(I) ARRAY   MRIZ(I),I=1, 6)

           1   1   1   1   1   1

        MORSE MEDIA IN INPUT ZONE(I) ARRAY  MMIZ(I),I=1, 6)
```

Reproduced from best available copy.

TABLE XX. (Cont'd.)



Reproduced from best available copy.

APPENDIX A

ENERGY INDEXING SCHEME

For most problems the user does not need to worry about the energy-indexing scheme used in the MØRSE Monte Carlo code. Energies corresponding to the group boundaries in the forward group structure are input and the code takes care of the rest. The analysis package SAMBØ performs the proper bookkeeping for labeling. A diagram for the energy group structure is given in Fig. A1. However, some clarification of the indexing scheme for several types of problems is advantageous. This clarification can perhaps best be made through the use of examples. Consider a five-group coupled neutron-gamma-ray problem with the following group boundaries:

Group	Energy (eV)
1	15(+6)
2	5(+6)
3	1(+6)
EBØTN	3(+3)
4	10(+6)
5	2(+6)
EBØTG	4(+5)

with NMGP = 3, NMTG = 5, NGPQTN = 3, NGPQTG = 2

There are several options which might be considered; namely, (1) a forward coupled problem, (2) a forward neutron-only problem, (3) a forward gamma-ray-only problem, (4) an adjoint coupled problem, (5) an adjoint neutron-only problem, and (6) an adjoint gamma-ray-only problem. Table AI gives the energies as they are indexed during a MØRSE run. Table AII gives the values of variables NQT1, NQT2, and NQT3 as they appear in USER common for the same six problems.

Preceding page blank

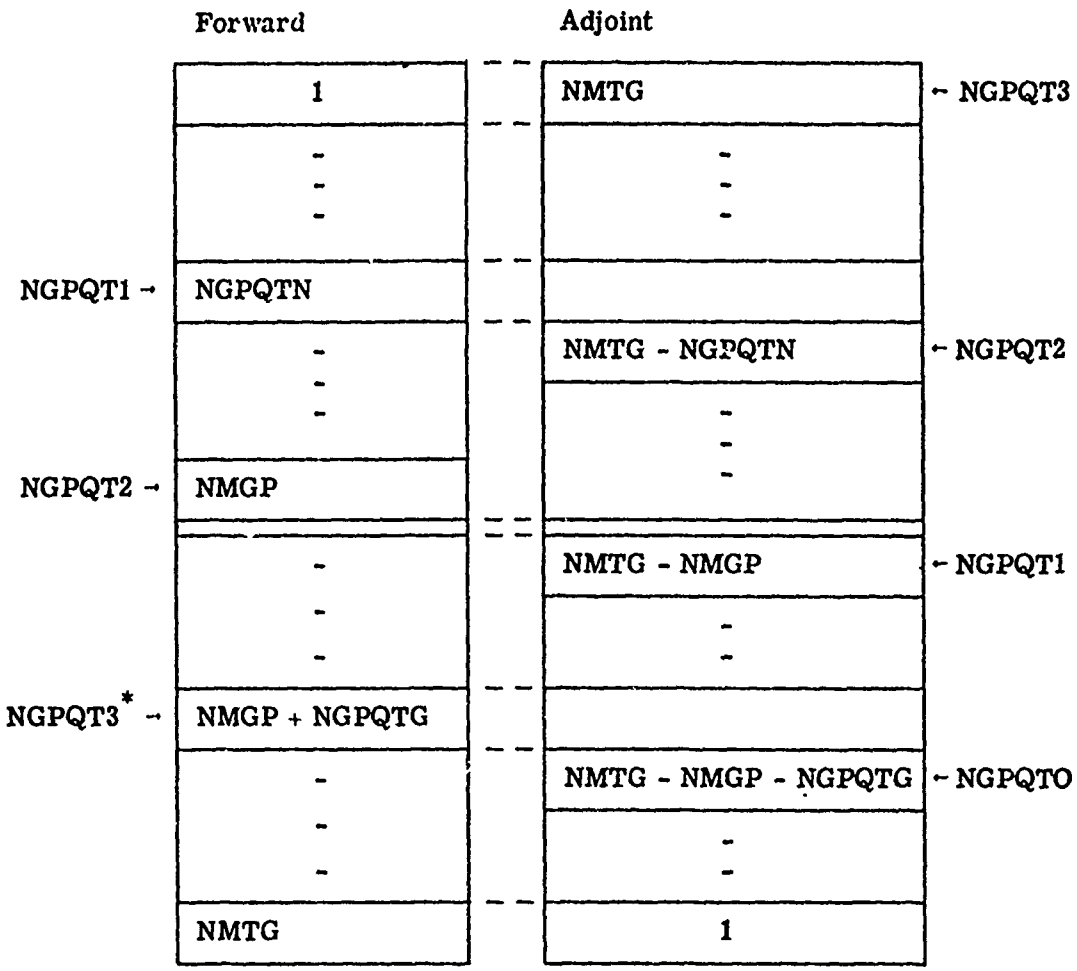


Fig. A1. Diagram of energy group structure.

* NGPQT3 = NGPQT1 for neutron only or gamma ray only problem.

TABLE AI.

Numerical Examples for Various Options with Corresponding Energies Given for Each Case

Group Index	N + γ Forward	N Forward	γ Forward	N + γ Adjoint	N Adjoint	γ Adjoint
1	15(+6)	15(+6)	--	2(+6)	1(+6)	2(+6)
2	5(+6)	5(+6)	--	10(+6)	5(+6)	10(+6)
3	1(+6)	1(+6)	--	1(+6)	15(+6)	--
4	10(+6)	--	10(+6)	5(+6)	--	--
5	2(+6)	--	2(+6)	15(+6)	--	--
EBØTN	3(+3)	3(+3)	--	3(+3)	3(+3)	--
EBØTG	4(+5)	--	4(+5)	4(+5)	--	4(+5)
NMGP	3	3	2	3	3	2
NMTG	5	3	2	5	3	2
NGPQTN*	3 2†	3 2	0 0	3 2	3 2	0 0
NGPQTG	2 1	0 0	2 1	2 1	0 0	2 1

* Note: A value of NGPQTN = 0 signals a gamma-ray-only problem in the random walk module. A value of NGP = 0 signals a gamma-ray-only problem in the cross-section module.

† Values are given for two cases with different values of NGPQTN and NGPQTG.

TABLE AI. (Cont'd.)

Group Index	N +γ Forward		N Forward		γ Forward		N +γ Adjoint		N Adjoint		γ Adjoint	
NQT1	3	2	3	2	2	1	2	2	0	0	2	2
NQT2	3	3	3	3	0	0	2	3	0	1	0	1
NQT3	5	4	3	2	2	1	5	5	3	3	2	2

TABLE AII.

Values of NQT1, NQT2, and NQT3 for Several Cases

Cases	NQT1=	NQT2=	NQT3=
Forward			
Combined	NGPQTN	NMGP	NMGP + NGPQTG
N Only	NGPQTN	NMGP	NQT1
γ Only	NGPQTG	NMGP*	NQT1
Adjoint			
Combined	NMTG - NMGP	NMTG - NGPQTN	NMTG
N Only	NMTG - NMGP	NMTG - NGPQTN	NMTG
γ Only	NMTG - NMGP*	NMTG - NGPQTG	NMTG

* NMGP is 0 here, but was > 0 in INPUT.

APPENDIX B

INPUT INSTRUCTIONS

B. 1 MØRSE Input Instructions

The input read† by subroutine INPUT is as follows:

CARD A (20A4)

Title card. (Any character other than a blank or alphameric in column one will terminate the job.)

CARD B (13I5)

NSTRT	-	number of particles per batch,
NMØST	-	maximum number of particles allowed for in the bank(s); may equal NSTRT if there is no splitting, fission, and secondary generation during execution. If bank size is exceeded by more than 50 due to fission or secondary gamma ray generation the job is terminated,
NITS	-	number of batches,
NQUIT	-	number of sets of NITS batches to be run without calling subroutine INPUT,
NGPQTN*	-	number of neutron groups being analyzed,
NGPQTG*	-	number of gamma-ray groups being analyzed,
NMGP*	-	number of primary particle groups for which cross sections are stored; should be same as NGP (or the same as NGG when NGP = 0) on Card XB read by subroutine XSEC,

† Input and output logical unit members are defined in MAIN and in G1. Unit 16 is a temporary storage unit used by JØMIN.

* See Table BI for sample input.

NMTG* - total number of groups for which cross sections are stored; should be same as NGP + NGG as read on Card XB read by subroutine XSEC,

NCØLTP - set greater than zero if a collision tape is desired; the collision tape is written by the user routine BANKR,

IADJM - set greater than zero for an adjoint problem,

MAXTIM - maximum clock time in minutes allowed for the problem to be on the computer,

MEDIA - number of cross-section media; should agree with NMED on Card XB read by subroutine XSEC,

MEDALB - albedo scattering medium is absolute value of MEDALB; if

= 0, no albedo information to be read in,

< 0, albedo only problem--no cross sections are to be read,

> 0, coupled albedo and transport problem.

CARD C (4I5, 5E10.5)

ISØUR - source energy group if > 0, if ISØUR ≤ 0, SØRIN is called for input of Cards E1 and E2,

NGPFS - number of groups for which the source spectrum is to be defined. If ISØUR ≤ 0, NGPFS ≥ 2,

ISBIAS - no source energy biasing if set equal to zero; otherwise the source energy is to be biased, and Cards E2 are required,

NØTUSD - an unused variable,

WTSTRT - weight assigned to each source particle,

EBØTN - lower energy limit of lowest neutron group (eV) (group NMGP),

EBØTG - lower energy limit of lowest gamma-ray group (eV) (group NMTG),

---

* See Table BI for sample input.

TCUT - age in sec at which particles are retired; if TCUT = 0, no time kill is performed,

VELTH - velocity of group NMGP when NGPQTN > 0; i.e., thermal-neutron velocity (cm/sec).

CARD D (7E10.4)

XSTRT, YSTRT, ZSTRT - coordinates for source particles,

AGSTRT - starting age for source particles,

UINP, VINP, WINP - source particle direction cosines; if all are zero, isotropic directions are chosen.

Source data on Cards C and D may be overridden by changes in subroutine SØURCE.

CARDS E1 (7E10.4) (omit if ISØUR on Card C > 0)

NGPFS values of FS, where FS equal the unnormalized fraction of source particles in each group.

CARDS E2 (7E10.4) (omit if ISØUR > 0 or if ISØUR ≤ 0 and ISBIAS = 0)

If ISBIAS > 0, NGPFS values of BFS, the relative importance of a source in group I, are required.

CARDS F (7E10.4)

NMTG values of ENER, the energies (in eV) at the upper edge of the energy group boundaries. (Note: The lower energies of groups NMGP and NMTG were read on Card C.)

CARD G (2I5, 5X, 36I1, 13I1) (omit if NCØLTP on Card B ≤ 0)

NHISTR - logical tape number for the first collision tape,

NHISMX - the highest logical number that a collision tape may be assigned,

NBIND(J), J=1, 36 - an index to indicate the collision parameters to be written on tape,

NCØLLS(J), J=1, 13 - an index to indicate the types of collisions to be put on tape. (See Table BII for information concerning NBIND's and NCØLLS.)

CARD H (Z12)

RANDØM - starting random number.

CARD I (7I5)

NSPLT - index indicating that splitting is allowed if > 0,

NKILL - index indicating that Russian roulette is allowed if > 0,

NPAST - index indicating that exponential transform is invoked if > 0 (subroutine DIREC required),

NØLEAK - index indicating that non-leakage is invoked if > 0,

IEBIAS - index indicating that energy biasing is allowed if > 0,

MXREG† - number of regions described by geometry input (will be set to one if ≤ 0),

MAXGP - group number of last group for which Russian roulette or splitting or exponential transform is to be performed.

† If ENDRUN is used, a data array relating media numbers to region numbers must be given in a data statement in ENDRUN.

CARD J (6I5, 4E10.5) (omit if NSPLT + NKILL + NPAST = 0)

NGP1, NDG, NGP2, NRG1, NDRG, NRG2 - from energy group NGP1 to energy group NGP2, inclusive, in steps of NDG and from region NRG1 to NRG2, inclusive, in steps of NDRG, the following weight standards and path-stretching parameters are assigned. If NGP1 = 0, groups 1 to MAXGP will be used; if NRG1 = 0, regions 1 to MXREG will be used (both in steps of one). Usually NDG = 1 and NDRG = 1.

WTHIH - weight above which splitting will occur,

WTLØW - weight below which Russian roulette is played,

WTAVE - weight given those particles surviving Russian roulette,

PATH - path-length stretching parameters for use in exponential transform (usually $0 \leq PATH < 1$).

The above information is repeated until data for all groups and regions are input.

End Cards J with negative value of NGP1 (ex., -1 in columns 4 and 5).

CARDS K (7E10.4) (omit if IEBIAS on Card I ≤ 0)

((EPRØB(IG, NREG), IG = 1, NMTG), NREG = 1, MXREG)

Values of the relative energy importance of particles leaving a collision in region NREG. Input for each region must start on a new card.

CARD L (4I5)

NSØUR - set ≤ 0 for a fixed source problem; otherwise the source is from fissions generated in a previous batch,

MFISTP - index for fission problem, if ≤ 0 no fissions are allowed,

NKCALC - the number of the first batch to be included in the estimate of k; if ≤ 0 no estimate of k is made,

NØRMF - the weight standards and fission weights are unchanged if ≤ 0; otherwise fission weights will be multiplied, at the end of each batch, by the latest estimate of k and the weight standards are multiplied by the ratio of fission weights produced in previous batch to the average starting weight for the previous batch. For time-dependent decaying systems, NØRMF should be > 0.

CARDS M (7E10.4) (omit if MFISTP on Card L ≤ 0)

(FWLØ(I), I = 1, MXREG) values of the weight to be assigned to fission neutrons.

CARDS N (7E10.4) (omit if MFISTP on Card L ≤ 0)

(FSE(IG, IMED), IG = 1, NMGP), IMED = 1, MEDIA) the fraction of fission-induced source particles in group IG and medium IMED.

Note: Input for each medium must start on a new card.

CARDS Ø (7E10.5) (omit if NGPQTN = 0 or NGPQTG = 0, i.e., include if coupled neutron-gamma-ray problem)

((GWLØ(IG, NREG), IG = 1, NMGP or NMTG - NMGP), NREG = 1, MXREG) - average number of secondary particles to be generated. NMGP groups are read for each region in a forward problem and NMTG - NMGP for an adjoint. Input for each region must start on a new card.

TABLE BI.

Sample Group Input Numbers for Some Representative Problems*

Case A - Neutron Only Cross Sections (22 groups)
Case B - Gamma-Ray Onlv Cross Sections (18 groups)
Case C - Neutron-Gamma-Ray-Coupled Cross Sections (22-18 groups)

Input Variable	Problem Type: Case A Top 14 groups	Case B Top 17 groups	Case C Neutrons Only Top 14 groups	Case C Gamma Rays Only Top 17 groups	Case C Neutron-Gamma Top 14 Neutron, Top 17 Gamma groups	
MØRSE Input:						
NGPQTN	14	0	14	0	14	
NGPQTG	0	17	0	17	17	CARD B,
NMGP	22	18	22	18	22	Variables
NMTG	22	18	22	18	40	
NGP	22	18	22	0	22	
NGG	0	0	0	18	18	CARD XB
INGP	22	18	40	40	40	Variables

* For cross sections with full downscatter, NDS = NGP, NDSG = NGG, INDS = INGP, and ITBL = number of downscatters + number of upscatters + 3. Usually, ISGG = number of upscatters + 4.

TABLE BII.

BANKR Arguments (NCØLLS)

BANKR Argument	Called from	Location of Call in Walk
-1	MØRSE	After call to INPUT--to set parameters for new problem
-2	MØRSE	At the beginning of each batch of NSTRT particles
-3	MØRSE	At the end of each batch of NSTRT particles
-4	MØRSE	At the end of each set of NITS batches--a new problem is about to begin
1	MSØUR	After a source event
2	TESTW	After a splitting has occurred--commented in column 1
3	FPRØB	After a fission has occurred
4	GSTØRE	After a secondary particle has been generated
5	MØRSE	After a real collision has occurred--post-collision parameters are available
6	MØRSE	After an albedo collision has occurred--post-collision parameters are available
7	NXTCØL	After a boundary crossing occurs (the track has encountered a new geometry medium other than the albedo or void media)
8	NXTCØL	After an escape occurs (the geometry has encountered medium zero)
9	MØRSE	After the post-collision energy group exceeds the maximum desired--commented in column 1

TABLE BII. (Cont'd.)

BANKR Argument	Called from	Location of Call in Walk
10	MØRSE	After the maximum chronological age has been exceeded--commented in column 1
11	TESTW	After a Russian roulette kill occurs--commented in column 1
12	TESTW	After a Russian roulette survival occurs--commented in column 1
13	GSTØRE	After a secondary particle has been generated but no room in the bank is available--commented in column 1.

B.2 Ø5R Geometry Input Instructions

Read by subroutine JØMIN and the specific input depends on the geometry packages used.

B.2.1 Spherical GEØM

CARD GA (I5, D10.5)

MED - medium number interior to R(> 0),

R - outer radius of sphere or spherical shell containing MED.

Repeat Card GA for all radii (≤ 20) in increasing order. End Card GA input with blank card.

CARD GB (D10.5)

R - region radius of sphere or spherical shell containing regions. Region numbers are assigned in consecutive order starting with 1, and R must be in increasing order.

Repeat Card GB for all radii (≤ 20) in increasing order. End Card GB input with blank card. If no regions are desired, a blank card must be used to signal no region geometry.

B.2.2 Slab GEØM

CARD GA (I5, D10.5)

MED - medium with Z as lower bound (> 0),

Z - lower limit of medium MED.

Repeat Card GA for all boundaries with the last card containing MED = 0 and the boundary of the system.

CARD GB (D10. 5)

Z - lower limit of region boundary. Region numbers are assigned in consecutive order starting with 1, and Z must be in increasing order.

Repeat Card GB for all region boundaries.

End Card GB input with a blank card. If no region geometry is desired, a blank card is required.

CARD GC (4D10. 5)

XL - lower boundary of system in X direction,

XU - upper boundary of system in X direction,

YL - lower boundary of system in Y direction,

YU - upper boundary of system in Y direction.

B. 2. 3 Cylindrical GEØM

CARD GA (I5, 5X, A8)

NREGIN - flag to indicate material media (=2) or both region and material media (=1),

SEX - sex of programmer.

CARD GB (E10. 5)

R - radii of the cylindrical shells describing the material media in ascending order.

Repeat Card GB until all radii have been input.

End Card GB input with a blank card.

CARD GC (E10.5, 12I5/8I5)

H - upper height of medium M(I) (> 0), cylinders assumed to start at H = 0,

M(I) - media for the cylindrical shells for this height.

Repeat Card GC until all height intervals have been input.

End Card GC input with a blank card or if there are more than 12 radial intervals, 2 blank cards.

CARD GC (E10.5) (omit if NREGIN = 2)

RG - radii of the cylindrical shells describing the region geometry in ascending order.

Repeat Card GD until all region geometry has been input.

End Card GD with a blank card.

CARD GE (E10.5, 12I5/8I5) (omit if NREGIN = 2)

HG - upper height of region MG(I),

MG(I) - region numbers for the cylindrical shells for this height.

Repeat Card GE until all height intervals have been input.

End Card GE input with a blank card or if there are more than 12 radial intervals, 2 blank cards.

B.2.4 General GEØM

CARD GA (I5, 5X, A6, 1X, A7) hollerith left adjusted

NSTAT - flag to indicate material media only if 2 and both region and material media if 1,

SEX - sex of the programmer (select one from MALE, FEMALE, or blank indicating uncertain),

STATUS - marital status of programmer.

CARD GB (2A4, A3, 5(D10.5, A1))

DUMMY(3) - hollerith characters not used,

FIN(I) - zone boundaries increasing order along the x axis,

BCD(I) - flag to indicate end of input if blank, comma means to continue.

Repeat in format (6(D10.5, A1))if more than five boundaries along the x axis are needed.

CARD GC - same as CARD GB except for y axis.

Repeat in format (6(D10.5, A1)) if more than five boundaries along the y axis are needed.

CARD GD - same as Card GB except for z axis.

Repeat in format (6(D10.5, A1)) if more than five boundaries along the z axis are needed.

CARD GE (A4, A2, 3I5)

BCD1 - hollerith ZØNE,

BCD2 - dummy,

NXZNØ, NYXNØ, NZZNØ - integers which specify the zone as being the NXZNØth zone in the x direction, NYZNØth zone in the y direction, and NZZNØth zone in the z direction.

CARD GF (2A4, A3, 5(D10.5, A1))

DUMMY(3) - hollerith characters not used.

FIN(I) - block boundaries in increasing order along the x axis,

BCD(I) - flag to indicate end of input if blank, comma means to continue.

Repeat in format (6(D10. 5, A1)) if more than five boundaries along the x axis are needed.

CARD GG - same as Card GF except for y axis.

Repeat in format (6(D10. 5, A1)) if more than five boundaries along the y axis are needed.

CARD GH - same as Card GF except for z axis.

Repeat in format (6(D10. 5, A1)) if more than five boundaries along the z axis are needed.

CARDS GI to GØ describe the geometry for a block and must be included for each block in the zone.

CARD GI (A4, A2, 3I5)

BCD1 - hollerith BLØC,

BCD2 - dummy,

NXBND, NYBND, NZBND - integers which specify the block as being the NXBNDth in the x direction, the NYBNDth in the y direction, and the NZBND in the z direction.

CARD GJ (3A4, 10(I5, A1))

NAM2 - hollerith MEDI,

DUM(2) - dummy,

INP(I) - a list of media sector by sector in the block,

BCD(I) - flag to indicate end of input if blank, a comma means to continue.

Continuation with 12(I5, A1).

CARD GK (3A4, 10(I5, A1))

NAM2 - hollerith SURF,

DUM(2) - dummy,

INP(I) - a list of quadratic surfaces appearing in the block Numbers must appear in the order the surfaces are described on Card GQ,

BCD(I) - flag to indicate end of input if blank, a comma means to continue.

Continuation of Card GK in 12(I5, A1) format is permissible.

CARD GL (A4, A2, 18I3)

S1 - hollerith SECT,

DUM - dummy,

IND(I) - the designation of each sector which describes the position of the sector relative to quadratic surfaces,

+1: sector is on positive side of surface,

-1: sector is on negative side of surface,

0: surface is not needed to define sector.

There must be a Card GL for each sector and references to quadratic surfaces must be in same order as they are listed on Card GQ.

CARD GM (3A4, 10(I5, A1)) (omit if NSTAT on Card GA = 2)

NAM2 - hollerith REGI,

DUM(2) - dummy,

INP(I) - a list of regions sector by sector in the block,

BCD(I) - flag to indicate end of input if blank, a comma means to continue.

Continuation with 12(I5, A1) format is permissible.

CARD GN (3A4, 10(I5, A1)) (omit if NSTAT on Card GA = 2)

NAM2	-	hollerith SURF,
DUM(2)	-	dummy,
INP(I) BCD(I)	-	same as for Card GK except for region input instead of material input.

CARD GO (A4, A2, 18I3) (omit if NSTAT on Card GA = 2)

S1	-	hollerith SECT,
DUM	-	dummy,
IND(I)	-	same as for Card GL except for region input instead of material input.

Repeat Cards GI to GØ for each block.

CARD GP (I5, 16A4, A2)

NØBD	-	total number of quadratic surfaces in the entire system,
DUM(I)	-	hollerith characters ignored by the code. (Helpful in identifying input at a later time.)

CARD GQ (4 (D10.5, A4, 1X, A1))

CØF(J)	-	coefficient of the term,
BCD1(J)	-	hollerith indicating which term of the equation. XSQ, YSQ, ZSQ, XZ, YX, YZ, XY, ZX, YZ, X, Y, Z, or blank are the possibilities,
BCD2(J)	-	a flag which indicates the quadratic equation continues. Any non-blank character ends the field. The next function must start on new card.

Repeat Cards GQ until all surfaces have been described.

A sample of the input is shown in Table BIII.

TABLE BIII.

Ø5R General Geometry Input Example

CODED BY _____

DATE _____ REQUEST NO. _____

FORTRAN

PROBLEM _____

PROGRAM _____

PAGE 1 OF 2

STATEMENT TYPE	STATEMENT NUMBER	CONT	FORTRAN STATEMENT	IDENTIFICATION
	1		MALE	CARD ØA
X	ZONE		-30.5, 30.5	CARD ØB
Y	ZONE		-30.5, 30.5	CARD ØC
Z	ZONE		0.0, 238.	CARD ØD
ZONE			1 1 1	CARD ØE
X	BLOCK		-30.5, 30.5	CARD ØF
Y	BLOCK		-30.5, 30.5	CARD ØG
Z	BLOCK		0.0, 14., 62.8, 182.5, 238.0	CARD ØH
BLOCK			1 1 1	CARD ØI
MEDIA			1000, 1	CARD ØJ
SURFACES			1	CARD ØK
SECTOR		-1		CARD ØL
SECTOR		+1		CARD ØL
BLOCK			1 2	CARD ØI
MEDIA			1000, 1	CARD ØJ
SURFACES			2	CARD ØK
SECTOR		-1		CARD ØL
SECTOR		+1		CARD ØL
BLOCK			1 3	CARD ØI
MEDIA			1000, 3 1	CARD ØJ

TABLE BIII. (Cont'd.)

FORTRAN

CODED BY _____

DATE _____ REQUEST NO. _____

PROBLEM _____

PROGRAM _____

PAGE 2 OF 2

STATEMENT TYPE	STATEMENT NUMBER	CONT	FORTRAN STATEMENT	IDENTIFICATION
			SURFACES 3, 4	CARD GK
			SECTOR -1 0	CARD GL
			SECTOR +1 -1	CARD GL
			SECTOR 0 +1	CARD GL
			BLOCK 1 1 4	CARD GJ
			MEDIA 1000, 1	CARD GJ
			SURFACES 5	CARD GK
			SECTOR -1	CARD GL
			SECTOR +1	CARD GL
			5EQUATIONS FOR DOUBLE TAPERED COLLIMATOR	CARD GS
			1. XSQ 1. YSQ -161.29 $	CARD GS
			1. XSQ 1. YSQ -.0223772 XSQ +4.42415 X	CARD GS
			-218.87 $	CARD GS
			1. XSQ 1. YSQ -29.160 $	CARD GS
			1. XSQ 1. YSQ -40.5225 $	CARD GS
			1. XSQ 1. YSQ -.0600933 ZSQ +19.5747 Z	CARD GS
			-1572.0868 $	

B.3 Combinatorial Geometry Input Instructions

The combinatorial geometry input data is read by the JØMIN subroutine, except for the region volumes VNØR(I), which are read by the GTVLIN subroutine whenever IVØPT = 3. For clarity of terminology, the terms "regions" and "media" have essentially the same meaning as in the Ø5R Geometry Package, but are constructed in a different manner. The term "zone" is the same as the "region" as defined in the original combinatorial geometry package. The term "body" has the same meaning as in the original combinatorial geometry package.

CARD CGA (2I5, 10X, 10A6)

IVØPT - option which defines the method by which region volumes are determined; if

IVØPT = 0, volumes set equal to 1.,

IVØPT = 1, concentric sphere volumes are calculated

IVØPT = 2, slab volumes (1-dim.) are calculated, *

IVØPT = 3, volumes are input by card,

IDBG - if IDBG > 0, subroutine PR is called to print results of combinatorial geometry calculations during execution. Use only for debugging,

JTY - alphanumeric title for geometry input (columns 21-80).

CARDS CGB (2X, A3, 1X, I4, 6E10.3)

One set of CGB cards is required for each body and for the END card (see Table B IV). Leave columns 1-6 blank on all continuation cards.

* Not operational.

ITYPE - specifies body type or END to terminate reading of body data (for example BØX, RPP, ARB, etc.). Leave blank for continuation cards,

IALP - body number assigned by user (all input body numbers must form a sequence set beginning at 1). If left blank, numbers are assigned sequentially. Either assign all or none of the numbers. Leave blank for continuation cards,

FPD(I) - real data required for the given body as shown in Table B IV.

CARDS CGC (2X, A3, I5, 9(A2, I5))

Input zone specification cards. One set of cards required for each input zone, with input zone numbers being assigned sequentially.

IALP - IALP must be a nonblank for the first card of each set of cards defining an input zone. If IALP is blank, this card is treated as a continuation of the previous zone card,

IALP = END denotes the end of zone description.

NAZ - total number of zones that can be entered upon leaving any of the bodies defined for this input region (some zones may be counted more than once). Leave blank for continuation cards for a given zone. (If NAZ ≤ 0on the first card of the zone card set, then it is set to 5). This is used to allocate blank common,

Alternate IIBIAS(I) and JTY(I) for all bodies defining this input zone.

IIBIAS(I) - specify the "ØR" operator if required for the JTY(I) body,

JTY(I) - body number with the (+) or (-) sign as required for the zone description

CARDS CGD (14I5)

MRIZ(I) - MRIZ(I) is the region number in which the "I^{th}" input zone is contained (I = 1, to the number of input zones). Region numbers must be sequentially defined from 1.

CARDS CGE (14I5)

MMIZ(I) - MMIZ(I) is the medium number in which the "I^{th}" input zone is contained (I = 1, to the number of input zones). Medium numbers must be sequentially defined from 1.

CARDS CGF (7E10.5) (omit if IVØPT $\neq$ 3)

VNØR(I) - volume of the "I^{th}" region (I = 1 to MXREG, the number of regions).

Note: If ENDRUN is used to obtain collision density and track length per unit volume estimate of fluence, then a data statement in ENDRUN must give a relationship between region and media. In this case only one medium may be in a region.

TABLE BIV.

Input Required on CGB Cards for Each Body Type

Card Columns Body Type	ITYPE 3-5	IALP 7-10	Real Data Defining Particular Body 11-20	21-30	31-40	41-50	51-60	61-70	Number of Cards Needed
Box	BØX	IALP is assigned by the user or by the code if left blank.	Vx	Vy	Vz	H1x	H1y	H1z	1 of 2
			H2x	H2y	H2z	H3x	H3y	H3z	2 of 2
Right Parallele-piped	RPP		Xmin	Xmax	Ymin	Ymax	Zmin	Zmax	1
Sphere	SPH		Vx	Vy	Vz	R	-	-	1
Right Circular Cylinder	RCC		Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
			R	-	-	-	-	-	2 of 2
Right Elliptic Cylinder	REC		Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
			R1x	R1y	R1z	R2x	R2y	R2z	2 of 2
Ellipsoid	ELL		V1x	V1y	V1z	V2x	V2y	V2z	1 of 2
			L	-	-	-	-	-	2 of 2
Truncated Right Cone	TRC		Vx	Vy	Vz	Hx	Hy	Hz	1 of 2
			L1	L2	-	-	-	-	2 of 2
Right Angle Wedge	WED		Vx	Vy	Vz	H1x	H1y	H1z	1 of 2
			H2x	H2y	H2z	H3x	H3y	H3z	2 of 2

TABLE BIV. (Cont'd.)

Card Columns Body Type	ITYPE 3-5	IALP 7-10	Real Data Defining Particular Body 11-20	21-30	31-40	41-50	51-60	61-70	Number of Cards Needed
Arbitrary Polyhedron	ARB		V1x	V1y	V1z	V2x	V2y	V2z	1 of 5
			V3x	V3y	V3z	V4x	V4y	V4z	2 of 5
			V5x	V5y	V5z	V6x	V6y	V6z	3 of 5
			V7x	V7y	V7z	V8x	V8y	V8z	4 of 5
			Face Descriptions (see note below)						5 of 5
Termination of Body Input Data	END								

Note: Card 5 of the arbitrary polyhedron input contains a four-digit integer for each of the six faces of an ARB body. The format is 6(1X, I4), beginning in column 11. See the ARB write-up in Section 2.1 for an example.

B.4 MØRSEC--Cross-Section Module Input Instructions

CARD XA (20A4)

Title card for cross sections. This title is also written on tape if a processed tape is written; therefore, it is suggested that the title be definitive.

CARD XB (13I5)

Column	Name		Description
5	NGP	-	the number of primary groups for which there are cross sections to be stored. Should be same as NMGP input in MØRSE,
10	NDS	-	number of primary downscatters for NGP (usually NGP),
15	NGG	-	number of secondary groups for which there are cross sections to be stored,
20	NDSG	-	number of secondary downscatters for NGG (usually NGG),
25	INGP	-	total number of groups for which cross sections are to be input,
30	ITBL	-	table length, i.e., the number of cross sections for each group (usually equal to number of downscatters + number of up-scatters + 3),
35	ISGG	-	location of within-group scattering cross sections (usually equal to number of up-scatters + 4),
40	NMED	-	number of media for which cross sections are to be stored--should be same as MEDIA input in MØRSE,
45	NELEM	-	number of elements for which cross sections are to be read,

CARD XB (Cont'd.)

Column	Name		Description
50	NMIX	-	number of mixing operations (elements times density operations) to be performed (must be ≥ 1),
55	NCØEF	-	number of coefficients for each element, including P_0,
60	NSCT	-	number of discrete angles (usually $NCØEF/2_{Integral}$),
65	ISTAT	-	flag to store Legendre coefficients if greater than zero.

CARD XC (11I5)

Column	Name		Description
5	IRDSG†	-	switch to print the cross sections as they are read if > 0, if < 0 card sequence is not checked,
10	ISTR†	-	switch to print cross sections as they are stored if > 0,
15	IFMU†	-	switch to print intermediate results of μ's calculation if > 0,
20	IMØM†	-	switch to print moments of angular distribution if > 0,
25	IPRIN†	-	switch to print angles and probabilities if > 0,
30	IPUN†	-	switch to print results of bad Legendre coefficients if > 0,
35	IDTF†	-	switch to signal that input format is DTF-IV format if > 0; otherwise, ANISN format is assumed,

† Switches are ignored if IXTAPE < 0.

CARD XC (Cont'd.)

Column	Name		Description
40	IXTAPE	-	logical tape unit if binary cross-section tape, set equal to 0 if cross sections are from cards. If negative, then the processed cross sections and other necessary data from a previous run will be read; in this case (IXTAPE < 0) no cross sections from cards and no mixing cards may be input. The absolute value of IXTAPE is the logical tape unit,
45	JXTAPE	-	logical tape unit of a processed cross-section tape to be written. This processed tape will contain the title card, the variables from common LØCSIG and the pertinent cross sections from blank common,
50	IØ6RT	-	logical tape unit of a point cross-section tape in Ø6R format,
55	IGQPT	-	last group (MØRSE multigroup structure) for which the Ø6R point cross sections are to be used (≤ NMGP).

CARD XD (14I5) (omit if IXTAPE ≤ 0)

Element identifiers for cross-section tape. If element identifiers are in same order as elements on tape, the efficiency of the code is increased due to fewer tape rewinds.

CARD XE (omit if IXTAPE ≠ 0)

Cross sections in ANISN format if IDTF ≤ 0, otherwise, DTF-IV format. Cross sections for INGP groups with a table length ITBL for NELEM elements each with NCØEF coefficients.

CARDS XF (2I5, E10.5) (NMIX cards are required, omit if IXTAPE < 0)

KM - medium number (media numbers from 1 to MEDIA, see Card B, must appear on some XF card),

KE - element number occurring in medium KM (negative value indicates last mixing operation for that medium and at least one negative value is required for each medium),

RHØ - density of element KE in medium KM in units of atoms/(barn cm).

CARDS XG (I5) (omit if IØ6RT ≤ 0)

NXPM - number of point cross-section sets per medium found on an Ø6R tape,

= 1, total cross section only,

= 2, total and scattering cross section,

= 3, total, scattering, and ν*fission cross section.

XCHEKR Card (4I5) Cross sections and cross-section input data may be checked independently of MØRSE utilizing XCHEKR.[5] The input to XCHEKR consists of the cross-section cards XA through XG preceded by a card as follows:

IADJM - set greater than zero for an adjoint problem,

MEDIA - number of cross-section media; should equal NMED on Card XB,

NMGP - number of primary particle energy groups for which cross sections are to be stored; should equal NGP (or NGG if NGP = 0) on Card XB,

NMTG - total number of energy groups for which cross section are to be stored. Should be equal to NGP + NGG on Card XB.

B.5 SAMBØ Analysis Input Instructions

The following data are read from cards by SCØRIN:

CARD AA (20A4)

Title information--will be immediately output.

CARD AB (8I5)

ND	-	number of detectors (set =1 if ≤ 0),
NNE	-	number of primary particle energy bins to be used, (must be ≤ NE),
NE	-	total number of energy bins (set =0 if ≤ 1),
NT	-	number of time bins for each detector (may be negative, in which case |NT| values are to be read and used for every detector) (set =0 if |NT| ≤ 1),
NA	-	number of angle bins (set =0 if ≤ 1),
NRESP	-	number of energy-dependent response functions to be used (set =1 if ≤ 0),
NEX	-	number of extra arrays of size NMTG to be set aside (useful, for example, as a place to store an array of group-to-group transfer probabilities for estimator routines. If the subroutine ENDRUN which outputs fluence estimates from collision and track lengths is used, then this number must be at least MXREG + 2 (see Card I for MXREG),
NEXND	-	number of extra arrays of size ND to be set aside (useful, for example, as a place to store detector-dependent counters).

CARDS AC (3E10.4) (ND cards will be read)

X, Y, Z - detector location. (If other than point detectors are desired, the point locations must still be input and can be combined with additional data built in to user routines to fully define each detector.)

Note that the distances between the above points and the XSTRT, YSTRT, ZSTRT values and the initial age, AGSTRT, will be used to define the lower limit of the first time bin.

CARD AD (20A4)

Title or units for total responses for all detectors. Will be used in columns 54 through 133 of the title for the print of these arrays.

CARD AE (20A4)

Title or units for each total response for all detectors.

CARDS AF (7E10.4)

Response function values. NMTG values will be read in each set of AF cards. Input order is from energy group 1 to NMTG (order of decreasing energy).

Note: Cards AE and AF are read in the following order: AE_1, $AF_1, \ldots, AF_n$, AE_2, $AF_1, \ldots, AF_n$, etc. NRESP sets of AE, AF cards will be read.

CARD AG (20A4) (omit if NE $\leq$ 1)

Units for energy-dependent fluence for all detectors.

CARDS AH (14I5) (omit if NE ≤ 1)

Energy group numbers defining lower limit of energy bins (in order of increasing group number). The NNE (if > 0) entry must equal NGPQTN; the NE entry must be set to NMGP + NGPQTG for a combined problem, or else NGPQTG or NGPQTN.

CARD AI (20A4) (omit if |NT| ≤ 1)

Units for time-dependent total responses for all detectors.

CARD AJ (20A4) (omit if |NT| ≤ 1 or NE ≤ 1)

Units for time and energy-dependent fluence for all detectors.

CARDS AK (7E10.4) (omit if |NT| ≤ 1)

NT values of upper limits of time bins for each detector (in order of increasing time and detector number). The values for each detector must start on a new card. |NT| values only are read if NT is negative. They are then used for every detector.

CARD AL (20A4) (omit if NA ≤ 1)

Units for angle- and energy-dependent fluence for all detectors.

CARD AM (7E10.4) (omit if NA ≤ 1)

NA values of upper limits of angle bins (actually cosine bins; the NA^{th} value must equal one).

Following the input for the SAMBØ analysis module, input cards for user-written routines INSCØR, SØURCE, and ENDRUN.

B.6 Input to PICTURE

1. Card PA: Format (I5)

   NUSE: The number of characters to read on Card B to replace the standard values of ATABLE. Leave Card A blank and omit Card B if the standard ATABLE is desired. NUSE ≤ 50.

2. Card PB: Format (50A1) (omit if NUSE = 0)

   ATABLE(I), I = 1, NUSE: The list of characters that are to be printed for each medium. For medium N, ATABLE (N+1) is printed. If N ≥ 47, ATABLE (48) is printed. The standard values of ATABLE are:

Medium Number	Character Printed
0 (external void)	
1 through 9	1 through 9
10 through 35	A through Z
36 through 46	various special characters
≥ 47 (including internal voids)	(blank)

3. GEØM input: Combinatorial geometry input.

4. Card PC: Format (2I2, 18A4)

   ICNT:
   - = 0 After this picture, return to Card PC for another picture with the same geometry.
   - = 1 After this picture, return to Card PA to read in a new GEØM input.

   IRG:
   - = -1 Display the region geometry.
   - = 0 Display the zone geometry.
   - = 1 Display the material geometry.

   TITLE(I), I = 1, 18: 72 characters to be printed as a title.

5. Card PD: Format (6E10.5)

X_{UL}, Y_{UL}, Z_{UL} — X, Y, and Z coordinates in the combinatorial geometry of the upper left corner of the picture.

X_{LR}, Y_{LR}, Z_{LR} — X, Y, and Z coordinates in the combinatorial geometry of the lower right corner of the picture.

NOTE: Card PD partially describes the plane of the slice by defining two points in the plane and designates the top, bottom, left and right sides of the picture.

6. Card PE: Format (6E10.5)

U_X, U_Y, U_Z — Direction numbers proportional to the direction cosines for the U axis of the picture. The U axis points down the printed page in the direction the page moves through the printer.

V_X, V_Y, V_Z — Direction numbers for the V axis of the picture. The V axis points to the right across the page.

NOTE: Card PE completes the description of the plane of the slice by giving a line in the plane, also specifies the orientation of the picture on the output.

7. Card PF: Format (2I5, 2E10.5)

NU: Number of intervals to print along the U axis (overrides DELU).

NV: Number of intervals to print along the V axis (overrides DELV).

DELU: Spacing (in GEØM units) of intervals along the U axis.

DELV: Spacing (in GEØM units) of intervals along the V axis.

NOTE: All four entries are not required as input on Card PF, see below for explanation.

Options:

1. $X_{LR} = Y_{LR} = Z_{LR} = 0.$

For this case NU and NV must be specified. In addition, either DELU or DELV must be specified. If the other is left blank, the code will produce an undistorted picture. If both DELU and DELV are specified the picture is likely to be distorted. The standard printers give 10 characters to the inch across a line but only 6 lines per inch down the page. Because of this DELV = .6*DELU is necessary to produce an undistorted picture.

2. X_{LR} or Y_{LR} or $Z_{LR} \neq 0.$

If any one variable on Card PD is specified, the code will calculate the others to produce an undistorted picture.

If both NU and DELU (or both NV and DELV) are specified, DELU (or DELV) will be ignored.

The U and V axes may have arbitrary orientation. (If they are not orthogonal, the resulting picture will be distorted.) In Option 1, the first point will be at $(X, Y, Z)_{UL}$, and the remaining points in the directions and at the distances specified. In Option 2, the range from X_{UL} to X_{LR} is divided into intervals and the calculated points will be at the midpoints of the intervals. The first point will be 1/2 interval past $(X, Y, Z)_{UL}$ and the final point will be within 1/2 interval of $(X, Y, Z)_{LR}$. If $(X, Y, Z)_{LR}$ does not lie on the U-V plane, or if the U and V axes are not orthogonal, the location of the final point is not readily predictable.

The simplest method to obtain the correct results is to specify two diagonal corners of the plane of the slice on Card PD, with the top having the short dimension, if it is not square, and the side having the long dimension. Then, on Card PE, specify the U axis to be parallel to the edge of the slice with the large dimension (left or

right side), and the V axis to be parallel to the edge of the slice with the small dimension (top or bottom). Finally, let the only entry on Card PF be NV equal to maximum number of characters per line on your printer, this will provide the largest undistorted picture.

REFERENCES

1. E. A. Straker, P. N. Stevens, D. C. Irving and V. R. Cain, "The MORSE Code - A Multigroup Neutron and Gamma Ray Monte Carlo Transport Code," ORNL-CF-70-2-31, 1970.

2. E. A. Straker, P. N. Stevens, D. C. Irving, and V. R. Cain, "The MORSE Code - A Multigroup Neutron and Gamma-Ray Monte Carlo Transport Code," ORNL-4585, 1970.

3. V. R. Cain, "SAMBO, A Collision Analysis Package for Monte Carlo Codes," ORNL-CF-70-9-1, 1970.

4. E. A. Straker and M. B. Emmett, "MORSEC, A Revised Cross-Section Module for the MORSE Multigroup Monte Carlo Code," ORNL-4716, 1971.

5. C. E. Burgart and E. A. Straker, "XCHEKR, A Multigroup Cross-Section Editing and Checking Code," ORNL-TM-3518, 1971.

6. C. E. Burgart, "MORSE-GEOM-MORSEC-SAMBO Input Instructions," ORNL-CF-71-8-22, 1971.

7. E. A. Straker and M. B. Emmett, "Collision Site Plotting Positions and Collision Density Fluence Estimates for the MORSE Monte Carlo Code," ORNL-TM-3585, 1971.

8. K. G. Adams, "MORSE-SAMBO CDC 6600 Package," Sandia Corporation (private communication).

9. W. Guber et al., "A Geometric Description Technique Suitable for Computer Analysis of Both the Nuclear and Conventional Vulnerability of Armored Military Vehicles," MAGI-6701, Mathematical Applications Group Inc., August 1967.

10. D. C. Irving and G. W. Morrison, "PICTURE: An Aid in Debugging GEØM Input Data," ORNL-TM-2892, 1970.

11. C. E. Burgart, Oak Ridge National Laboratory, private communication (1971).

12. W. Guber, Mathematical Applications Group, Inc., private communication, 1971.